

Edgardo Richard Ventura (Eddie)  
 Frank Alvino  
 CIS121 061 - Introduction to Programming  
 4/6/2024

CIS 121 Introduction to Programming  
 Problem Set 10 – Pass By Reference

1. Allow the user to enter a quantity and price, use `ctrl+z` to stop. Use one function to compute the total (quantity times price), tax (7% of total) and total order (total plus tax). The function should be passed the quantity and price by value and total, tax and total order by reference. Display total, tax and total order in main. Sum and display total of all orders and tax for all orders and display after the loop (all data is processed).

<i>Input</i>	<i>Process</i>	<i>Output</i>
q		T
p		Tt
		To
	Cumulatively update totals for all orders before user presses <code>Ctrl+Z</code> : <code>gt += t, gtt += tt, gto += to</code>	Gt
		gtt
		count

<i>Name</i>	<i>Etymology</i>
Q	quantity
P	price
T	total before tax
Tt	total tax
To	total order with tax
Gt	grand total of all orders
gtt	grand total tax of all orders
count	Accumulation of items

2. Enter the weight of a package and zip code. Use `ctrl+z` to stop. Use a single function to do the computations specified next. Pass these weight and zip code by value, and pass postage, area charge and weight charge by reference. Compute postage to be sum of weight charge and area charge. Use tables below to find the charges. Compute weight charge to be weight x weight charge per ounce. Find the area charge in the table based on zip code. Then compute postage to be area charge plus weight charge. The function should return the weight charge, area charge and postage. Display area charge, weight charge and postage. Count and display the number of entries made.

Area Table – Used to determine the area charge	
<u>Area</u>	<u>Area Charge</u>
60171	\$2.00
60172	\$2.50
60635	\$3.00
All others	\$5.00

Weight Table – used to determine the weight charge	
<u>Weight</u>	<u>Weight Charge per Ounce</u>
>100	0.02
>50	0.03
All other	0.05

<i>Input</i>	<i>Process</i>	<i>Output</i>
Weight		Postage
Zipcodeinput		Areacharge
		Weightcharge
	Update total postage and count of packages before user presses Ctrl+Z: totalPostage += postage, count++	Count totalpostage

<i>Name</i>
Weight
Areacharge
Weightcharge
Postage
Zipcodeinput
Zipcode
Count
Total postage

3. Enter the student's last name, credit hours and financial aid, use `ctl+z` to stop. Pass credit hours and financial aid to a function by value. Pass tuition and tuition owed by reference. Compute tuition to be credit hours times \$250. Compute tuition owed to be tuition minus the financial aid. Display student's last name, tuition and tuition owed. Sum and display total tuition owed by all students, count of number of entries and average amount owed by students.

<i>Input</i>	<i>Process</i>	<i>Output</i>
		Name
		Hrs
Name		Cost
		Aid
Hrs		Owed
aide		
	Update total tuition owed and count of students before user presses Ctrl+Z: totalOwed += owed, count++	
		Count totalowed

<i>Name</i>	<i>Etymology</i>
Name	Student last name
Hrs	Credit hours taken
Aid	Financial aid amount
Cost	Tuition cost
Owed	Tuition owed after aid
totalowed	Total tuition owed
count	Number of students

4. Enter a number of widgets, use `ctl+z` to stop. Pass the number to a function by value, use `ctl+z` to stop. Use a single function to determine the cost per widget using the cost table below. Then compute extended price (number of widgets x cost per widget) and 7% sales tax. Finally compute total order to be extended price plus sales tax. Pass cost per widget, extended price, sales tax and total order by reference. For each line, display number of widgets, cost per widget, extended price, sales tax and total order. Sum all total orders and display when there is no more data to process.

<i>Input</i>	<i>Process</i>	<i>Output</i>
		Widget
widget		Costperwidget
		Extendedprice
		Salestax
		Totalorder
	Update grand total of all orders before user presses Ctrl+Z: grandTotal += totalOrder	grandtotal

<i>Name</i>
Widget
Costperwidget
Extendedprice
Salestax
Totalorder
grandtotal

5. Enter the amount of investment, the 5 year interest rate and 10 year interest rate, use `ctrl+z` to stop. Pass the amount and interest rates to a function by value. Pass variables representing five year amount and ten year amount to the same function by reference. Compute the five year amount and ten year amount using the formula below. Display the amount of the investment, the five year amount and the ten year amount.

Five year amount = amount of the investment x (1 + 5 year rate) raised to 5<sup>th</sup> power.

Ten year amount = amount of the investment x (1+10 year rate) raised to 10<sup>th</sup> power.

Note: Enter the 5 year rate and 10 year rate in decimal form, i.e 5% is entered as 0.05.

Also Note: you need to use the `pow` built in function. Recall the `pow` function syntax:

`pow (base, exponent)`

In this case, base is 1 + 5 year rate and exponent is 5.

Another line will be: base is 1 + 10 year rate and exponent is 10.

Use `#include<math.h>` for the `pow` function.

<i>Input</i>	<i>Process</i>	<i>Output</i>
amt		
R5	Calculate value after 10 years: <code>amt10 = amt * pow((1 + r10), 10)</code>	Amt
R10		Amt5 Amt10

<i>Name</i>	<i>Etymology</i>
Amt	initial investment amount
R5	5-year interest rate
R10	10-year interest rate
Amt5	value after 5 years
Amt10	value after 10 years