Edgardo Richard Ventura (Eddie)
Frank Alvino
CIS121 061 - Introduction to Programming
3/2/2024

CIS 121 Introduction to Programming
Problem Set 6 – Nested If and compound relational conditions.

1. The input to the problem is quantity of widgets and customer status. You set the price based upon quantity and status using the table below. Your program should determine the price to charge based on the schedule below. Calculate the extended price. Calculate tax at 7%. Display the extended price, tax amount and total.

Quantity	Status	Price
>10000	A	$10
>10000	B	$12
5000 to 10000	C	$20
5000 to 10000	D	$22
Below 5000	Any	$30

Input	*Process*	*Output*

	`if (quantity > 10000) {`
	`  if (stat == 'A' || stat == 'a') price = 10;`
	`  else if (stat == 'B' || stat == 'b') price = 12;`
	`}`	

quantity		exprice

stat	`else if (quantity >= 5000 && quantity <= 10000) {`
	`  if (stat == 'C' || stat == 'c') price = 20;`
	`  else if (stat == 'D' || stat == 'd') price = 22;`
	`}`
	`else if (quantity < 5000)`
	`  price = 30;`
		tax

	`exprice = quantity * price;`	total
	`tax = exprice * 0.07;`	
	`total = exprice + tax;`	

Name	*Etymology*
quantity	Quantity
stat	Customer Status
price	Price
exprice	Extended Price
tax	Tax
total	Total

2. Enter a part number of the following (10, 99, 55, 70, 50). Also enter the quantity. Determine the cost per unit using the table below. Then calculate the total cost. Display the part number, cost per unit and total.

Part	Quantity	Cost Per Unit
10	>1000	1.00
99	>500	2.00
All Others	All Others	5.00

Input	*Process*	*Output*
	`if (pn == 10 && quantity > 1000)` `  cpu = 1.00;`	
pn	`else if (pn == 99 && quantity > 500)` `  cpu = 2.00;`	pn
quantity	`else` `  cpu = 5.00;`	cpu
	`total = quantity * cpu;`	total

Name	*Etymology*
pn	Part Number
quantity	Quantity
cpu	Cost Per Unit
total	Total Cost

3. Allow the user to enter the number of concert tickets and location code (H, L). The price per ticket depends on the volume and location (see below). Display the number of tickets, price per ticket and the total cost.

   Volume is greater than 25 or location is H cost per ticket is $30.00
   Volume is greater than 10 and is equal to or less than 24, or the location is L cost per ticket is $40.00
   All other quantities or locations are $50.00

Input	*Process*	*Output*
	`if (nt > 25 || lc == 'H') ppt = 30.00;`	
Nt	`else if (nt > 10 || lc == 'L') ppt = 40.00;`	Nt
lc	`else ppt = 50.00;`	Ppt
	`total = nt * ppt;`	total

Name	*Etymology*
nt	Number of Tickets
lc	Location Code
Ppt	Price Per Ticket
total	Total Cost

4. Allow the user to enter the equipment code of a rental and a code indicating half day or full day. Determine the cost of the rental. Display the rental cost.

Equipment Code	Day	Cost
A	F	10.00
A	H	15.00
B	F	20.00
B	H	35.00
C	H	40.00
C	F	45.00
All others	All others	50.00

Input	*Process*	*Output*

	`if (ec == 'A') {`
	`  if (dc == 'F')`
	`  rc = 10.00;`
	`  else`
	`  rc = 15.00;`
	`}`
	`else if (ec == 'B') {`
	`  if (dc == 'F')`
	`  rc = 20.00;`
	`  else`
	`  rc = 35.00;`
	`}`
	`else if (ec == 'C') {`
	`  if (dc == 'F')`
	`  rc = 45.00;`
	`  else`
	`  rc = 40.00;`
	`  }`
	`else`
	`  rc = 50.00;`	

ec		rc
dc		

Name	*Etymology*
ec	Equi Code
dc	Day Code
rc	Rental Cost

5. You need to display the gross salary for an employee. They input a job code and Hours. First, determine the rate of pay based on job code and hours (see table below). Next, compute gross pay (hrs * rate). No overtime pay.

Job Code	Hours	Rate of Pay
L	> 40	50.00
L	<=40	40.00
J	>60	100.00
J	<=60	75.00
A	>40	25.00
A	<=40	20.00

Input	***Process***	***Output***
	`if (jobcode == 'l') {`	
`jc`	`    if (hours > 40) rateofpay = 50.00;`	
	`    else rateofpay = 40.00;`	
	`}`	
	`else if (jobcode == 'j') {`	
	`    if (hours > 60) rateofpay = 100.00;`	
	`    else rateofpay = 75.00;`	
`hours`	`}`	`gross`
	`else if (jobcode == 'a') {`	
	`    if (hours > 40) rateofpay = 25.00;`	
	`    else rateofpay = 20.00;`	
	`}`	
	`grosssalary = hours * rateofpay;`	

Name	***Etymology***
`jc`	`Job Code`
`hours`	`Hours`
`rop`	`Rate of Pay`
`gross`	`Gross Salary`