CIS 106

Problems – Using Nested if and Compound Relational Conditions

For each problem, develop the IPO and Code.

1. The input to the problem is quantity of widgets. Your program should determine the price to charge based on the schedule below. Calculate the extended price (quantity x price). Calculate tax at 7%. Display the extended price, tax amount and total.

Quantity Price

>10000 $10

5000 to 10000 $20

Below 5000 $30

|  |  |  |
| --- | --- | --- |
| Inputs | Processes | Outputs |
|  |  |  |
| qty | if qty > 10000  price = 10.00  else if qty >= 5000  price = 20.00  else  price = 30.00 | extendedprice |
|  | Extendedprice = qty \* price | taxamount |
|  | Taxamount = 0.07 \* extendedprice | total |
|  | Total = extendedprice + tax |  |
|  |  |  |
|  |  |  |

1. Enter a part number and quantity Determine the cost per unit using the table below. Then calculate the total cost (quantity x unit cost). Display the part number, cost per unit and total cost. Note: Part number can be an integer but it can also be a string because you are not doing arithmetic on it. However in your code if statement be sure to compare using consistency, that is, if item == “10” when item is a string and if item == 10 when item is an integer.

Part Unit Cost

10 **or** 55 1.00

99 2.00

80 **or** 70 3.00

All others 5.00

|  |  |  |
| --- | --- | --- |
| Inputs | Processes | Outputs |
|  |  |  |
| Part | if part == “10” OR part == “55”  unitcost = 1.00  else if part == “99”  unitcost = 2.00  else if part == “80” OR part == “70”  unitcost = 3.00  else  unitcost = 5.00 | Partnumber |
| qty | Total = qty \* unitcost | Unitcost |
|  |  | total |
|  |  |  |
|  |  |  |
|  |  |  |

1. Enter a principle amount of a CD and year to maturity of CD. Determine the interest rate based on the amount of the principle **and** maturity (see below). Calculate first year interest (principle x interest rate). Display principle, interest rate and the interest amount for first year.

Principle Years to Maturity Interest Rate

>$100,000 5 6%

$50,000 to $100,000 10 5%

$50,000 to $100,000 5 4%

Any other principle and years 2%

|  |  |  |
| --- | --- | --- |
| Inputs | Processes | Outputs |
|  |  |  |
| Principle | If principle > 100000 AND year = “5”  Rate = 0.06  Else if principle >= 50000 AND year = “10”  Rate = 0.05  Else if principle >= 50000 AND year = “5”  Rate = 0.04  Else  Rate = 0.02 | Principle |
| year | Interestamount = principle \* rate | Rate |
|  |  | interestamount |
|  |  |  |
|  |  |  |
|  |  |  |

1. Allow the user to enter number of concert tickets. The price per ticket depends on the volume (see below). Display the number of tickets, price per ticket and the total cost (number of tickets x Price Per Ticket).

Quantity Price Per Ticket

>=25 $50

10 to 24 $60

5 to 9 $70

Less 5 $75

|  |  |  |
| --- | --- | --- |
| Inputs | Processes | Outputs |
|  |  |  |
| qtytickets | If qtytickets >= 25  Price = 50.00  Else if qtytickets >=10  Price = 60.00  Else if qtytickets >= 5  Price = 70.00  Else  Price = 75.00 | qtytickets |
|  | Total = qtytickets \* price | Price |
|  |  | total |
|  |  |  |
|  |  |  |
|  |  |  |

1. The user will enter employee last name, salary and job level (as noted below). Use the job level to determine the bonus rate. Then compute bonus to be salary times bonus rate. Display employee last name and bonus.

Job Level Bonus Rate

10 and above 25%

5 to 9 20%

All others 10%

|  |  |  |
| --- | --- | --- |
| Inputs | Processes | Outputs |
|  |  |  |
| Lname | If level => 10  Rate = 0.25  Else if level >=5  Rate = 0.20  Else  Rate = 0.10 | Lname |
| Salary | Bonus = salary \* rate | bonus |
| level |  |  |
|  |  |  |
|  |  |  |
|  |  |  |