Problem Set – Introduction to Functions.

1. Allow the user to repeatedly enter a quantity and price. Prompt the user on whether they want to do the program (Yes or No). Use a function to compute the total (quantity times price). The function should be passed the quantity and price and then return the total. In the function, provide a 10% discount if the total is over $10,0000.00. Display quantity, price and total. Sum and display the extended price.

| Input | Process | Output |
|---|---|---|
| | CompExtPrice(qty, unitprice<br>Extprice = qty * unitprice<br><br>If extprice > 10000<br>Discount = extprice * 0.10<br>Else<br>Discamt = 0<br><br>Newextprice = extprice – discamt<br><br>Return newextprice | Extprice |
| Qty | | totalextprice |
| price | Main<br>totalExtPrice = 0<br><br>do you want to do this program? (Yes/No)<br><br>while (yes)<br>input qty, price<br>extprice = compextprice(qty, price)<br>display extprice<br>totalextprice = totalextprice + extprice<br>do you want to continue this program? | |
| | Display totalextprice | |
| | | |
| | | |
| | | |

2. Enter players last name, number of hits and at bats at the keyboard. Prompt the user on whether they want to do the program (Yes or No). Use a function to compute batting average. Pass the hits and at bats to the function. The function should return batting average. Display last name and batting average. Give a count of the number of players entered.

| Input | Process | Output |
|---|---|---|
| Lname | CompBatAvg(numHits, atBats)<br>batAvg = numHits / atBats | Lname, batAvg |

| | | |
|---|---|---|
| | return batAvg | |
| numHits | Main<br>C = 0<br><br>Prompt user (y/n)<br><br>While (yes)<br>Input lname, numHits, atBats<br>batAvg = compbatavg(numhits, atbats)<br>display lname, batAvg<br>c = c + 1<br>promp user again<br><br>display c | c |
| atBats | | |
| | | |
| | | |
| | | |
| | | |

3. Enter the destination city, miles travelled and gallons used for a trip. Prompt the user on whether they want to do the program (Yes or No). Use a function to compute miles per gallon. Pass miles travelled and gallons used to the function. The function should return miles per gallon. Count the number of entries made (number of trips) Display destination city, miles and mpg. At end display the number of entries made.

| Input | Process | Output |
|---|---|---|
| | CompMPG(miles, gallons)<br>Mpg = miles / gallons<br>Return mpg | City, miles, mpg |
| City | main<br>C = 0<br><br>Prompt user (y/n)<br><br>While (yes)<br>Input city, miles, gallons<br>Mpg = compmpg(miles, gallons)<br>Display city, miles, mpg<br>C = c + 1<br>Promp user again<br><br>Display c | c |
| Miles | | |
| gallons | | |

| | | |
|---|---|---|
| | | |
| | | |

4. Allow the employee to enter last name, job code and hours worked. Prompt the user on whether they want to do the program (Yes or No). Use a function to determine the pay rate. Pass to this function the job code and it should return rate of pay. Use Job code L is $25/hr, A is $30/hr and J is $50/hr for respective pay rates. Compute gross pay. Give time and a half for overtime. Display last name and gross pay. Sum and display total of all gross pay.

| Input | Process | Output |
|---|---|---|
| Lname | CompPayRate(jobcode)<br>If jobcode == "L"<br>Payrate = 25<br>If jobcode == "A"<br>Payrate = 30<br>If jobcode == "J"<br>Payrate = 50<br>Return payrate | Lname, gross pay |
| Jobcode | | totalpay |
| hours | Main<br>Totalpay = 0<br><br>Prompt user<br><br>While (yes)<br>Input lname, jobcode, hours<br>Payrate = CompPayRate(jobcode)<br>grosspay = payrate * hours<br>If hours > 40<br>Grosspay = grosspay + (hours – 40) /2 *payrate<br>Totalpay = totalpay + grosspay<br>Display lname, gross pay<br>Prompt user again | |
| | | |
| | | |
| | | |
| | | |

5. Allow the user to enter student last name, credit hours and district code. Prompt the user on whether they want to do the program (Yes or No). Use a function to compute tuition owed.

Charge In district (code of I)  $250 per credit hour. Out of district (code of O) is $550 per credit hour. The function should receive credit hours and district code and return tuition owed. Display student name and tuition owed. Sum and display total of all tuition owed.

| Input | Process | Output |
|---|---|---|
| Lname | Comptuition(dcode, hours)<br>If dcode = "I"<br>Rate = 250<br>Else<br>Rate = 550<br>Tuition = rate * hours<br>Return tuition | Lanme, tuition |
| Hours | Main<br><br>Tuitionsum = 0.00<br><br>Prompt user<br>While (yes)<br>Input lname, hours, dcode<br>Tuition = Comptuition(dcode, hours)<br>Tuitionsum = tuitionsum + tuition<br>Display lname, tuition | Tuitionsum |
| dcode | Display tuitionsum | |
| | | |
| | | |
| | | |
| | | |