Problem Set – While Loops. Develop IPO for each of the problems below and then save within this document. Then Write the code files using C++. Upload both IPO and .CPP files to Blackboard.

1. Allow any number of users to enter a quantity and price at the keyboard (use ctl+z to stop). Compute the extended price (quantity times price). If quantity is over 1000 give a 10% discount. Display quantity, price and extended, discount amount and discounted price for each entry. Keep a sum of the total for all the discounted prices. Display the total of discounted prices after all entries have been entered.

|  |  |  |
| --- | --- | --- |
| INPUT | PROCESS | OUTPUT |
| Quantity  Price | While loop  Extended price : quantity x price  if quantity > 1000  discount = extended x 10%  else  discount = 0%  calculate discounted price: extended – discount  total of all discounted prices | Quantity, Price, Extended Price  Discount Amount, Discounted Price  Total of all discounted prices |

1. Allow any number of players to enter last name, number of hits and at bats at the keyboard (use ctl+z to stop). Compute batting average (hits/ at bats). Display last name and batting average for each player. Keep a count of the number of players (or entries) made. Display the count after all entries have been made.

|  |  |  |
| --- | --- | --- |
| INPUT | PROCESS | OUTPUT |
| Last name  Number of hits  Number of at bats | Divide hits by at bats to compute batting average  At bats is not equal to 0 to avoid dividing by 0  Count each player entry  Loop continues until ctrl+z | Lastname and batting average  Total number of players entered |

1. Enter destination city, miles travelled to get there and gallons of gasoline used for any numberof trips entered at the keyboard (use ctl+z to stop). Compute miles per gallon (miles travelled / gallons used). Display the destination city and miles per gallon for each trip entered. Sum the miles travelled and give a count of the number of trips made. Display these at the end of the program.

|  |  |  |
| --- | --- | --- |
| INPUT | PROCESS | OUTPUT |
| Destination city  Miles traveled  Gallons of gasoline used | Miles per gallon = miles/ gallon  Total miles  Count each trip entry  Loop continues | Destination city and miles per gallon for each trip  Total miles traveled  Total number of trips made |

1. Allow the employee to enter last name, job code and hours worked (use ctl+z to stop). Calculate pay. (Job code L is $25/hr, A is $30/hr and J is $50/hr). Give time and a half for overtime. Display last name, job code, hours worked and pay for each employee. Sum the pay for each employee as well as count the entries made. After all entries are made, compute and display the average pay and the number of entries made.

|  |  |  |
| --- | --- | --- |
| INPUT | PROCESS | OUTPUT |
| Last name  Job code  Hours worked | Determine hourly rate based on job code: L = $25, A = $30, J = $50  If hours > 40, compute overtime at 1.5× rate for hours beyond 40  Hours worked (float)  Calculate regular and overtime pay  Accumulate total pay and count of employees  Loop continues until end-of-file (Ctrl+Z) | Display job code, last name, hours worked, and total pay for each employee  Total number of employees entered  Average pay across all employees |

1. Allow the user to enter student last name, credit hours and district code for any number of students (use ctl+z to stop). Compute tuition owed. In district (code of I) is charged $250 per credit hour. Out of district (code of O) is $550 per credit hour. Display student name and tuition owed for each entry. Sum the amount of tuition owed for all students as well and the total credit hours taken and finally the number of students who entered data. Display total tuition, total credit hours taken and count of number of students at the end.

|  |  |  |
| --- | --- | --- |
| INPUT | PROCESS | OUTPUT |
| Student last name  Credit hours  District code | Determine rate:  250/credit for in district  550/ credit for out of district  Multiply credit hours x rate = tuition  Accumulate total tuition and total credit hours  Count each entry | Student name and tuition owed  Total tuition for all students  Total credit hours taken  Total number of students entered |