CIS 106 – Loops Part 2

For each problem prepare an IPO chart. Then write the code for each. Save the IPO within this document and upload to your repository. After code is complete upload the files (.py) to your repository. Paste the link to your repository into the assignment completion link in Blackboard.

1. Allow the user to enter a principle amount and interest rate repeatedly (need a loop to control the program execution). Compute the annual interest (principle x rate). Compute ending balance to be principle (beginning balance + interest). Display year, beginning balance and ending balance for each of the 5 years. Display the accumulated interest for the 5 years. Note: the new balance by year (this will be the principle for the following year. Format the output.

|  |  |  |
| --- | --- | --- |
| Input | Process | Output |
|  | Total\_Interest = 0  Year = 0 |  |
| principle\_amount | Repeat process for 5 years  Year = year + 1 | Year  Beginning Balance  Ending Balance |
| interest | Annual\_interest = principle \* rate  Ending\_Balance = principle + interest  Total\_Interest = Total\_Interest + Annual\_Interest | Accumulated interest over 5 years |
|  | Display year, beginning\_balance, Ending\_Balance |  |
|  | Display Total\_Interest |  |

1. Fibonacci sequence is a sequence of natural order. The sequence is:

1, 1, 2, 3, 5, 8 etc

Use of for loop compute and display first 20 numbers in the sequence. Hint: start with 1 , 1.

|  |  |  |
| --- | --- | --- |
| Input | Process | Output |
| None | First = 1  Second = 1 | Fibonacci Sequence for up to 20 numbers |
|  | For num\_numbers  Next\_number = first + second  First = Second  Second = Next\_number |  |
|  |  |  |
|  | Display each next\_number in sequence |  |

1. Create a text file that contains employee last name and salary. Read in this data. Determine the bonus rate based on the chart below. Use that rate to compute bonus. For each line display the employee last name, salary and bonus. After the loop display the sum of all bonuses paid out.

|  |  |  |
| --- | --- | --- |
| Input | Process | Output |
|  | Totalbonus = 0 |  |
| Employee | Get last\_name | Last Name |
| salary | Get salary  If salary >= 100000  Bonus = salary \* .20  Else salary >= 50000  Bonus = salary \* .15  Else  Bonus = salary \* .10 | Salary  Bonus |
|  |  |  |
|  | Totalbonus= bonus + Totalbonus | Total Bonus |
|  | Display Totalbonus |  |

1. Create a text file with item, quantity and price. Read through the file one line at a time. Compute the extended price (quantity x price). For each line display the item, quantity, price and extended price. After the loop display the sum of all the extended prices, the count of the number of orders and the average order.

|  |  |  |
| --- | --- | --- |
| Input | Process | Output |
|  |  |  |
| item | C = 0  Tot\_ep = 0 |  |
| qty | Get item  While item !=””  Get qty, price  Ep = qty \* price  C = c+ 1  Tot\_ep = tot\_ep + ep  Display item, qty, price, ep  Get next item | Item  Price  Qty  Ep |
| price |  |  |
|  | Avg = tot\_ep / c | C  Tot\_ep  Avg |
|  | Display c, tot\_ep, avg |  |

1. Create a text file with student last name, district code (I or O) and number of credits taken. Compute tuition owed (credits taken x cost per credit). Cost per credit for in district students (district code I) is 250.00. Out of district students pay 500.00 per credit. For each line display student last name, credits taken and tuition owed. After the loop display sum of all tuition owed and the number of students.

|  |  |  |
| --- | --- | --- |
| Input | Process | Output |
|  | Numberofstudents = 0 |  |
| Studentname | Get lastname | Last Name |
| credits | Get creditstaken  Tuition = creditstaken \* costpercredit  IF district == “I”  Costpercredit = 250.00  ELIF district == “O”  Costpercredit = 500.00 | Credits Taken  Tuition Owed |
|  |  |  |
|  | Numberofstudents= numberofstudents + 1 | All Tuition Owed  Number of Students |