Problem 1

Input:

Principal amount (float)

Interest rate (float)

Process: I

initialize accumulated interest to 0

For each year (1 through 5):

Compute annual interest (principal \* rate)

Add annual interest to accumulated interest

Compute ending balance (principal + annual interest)

Set new beginning balance for the next year to the current ending balance

Print year, beginning balance, and ending balance

Output:

Table with columns for Year, Beginning Balance, and Ending Balance for each of the 5 years

Total accumulated interest over the 5 years

Problem 2

Input

The number of terms to display (set to 20).

Process:

Start the first two numbers of the Fibonacci sequence. Use a loop to calculate each number by adding the previous two numbers. Repeat this process until 20 numbers are generated.

Output:

The first 20 numbers of the Fibonacci sequence

Problem 3

Input:

Text file with employee last names and salaries.

Process:

Read the employee data from the file. Determine the bonus rate based on the salary. Calculate the bonus amount (salary \* bonus rate). Keep a running total of all bonuses.

Output: for each employee, display the last name, salary, and bonus. Total of all bonuses paid out.

Problem 4

Input:

Text file with items, quantities, and prices.

Process: Read the item data from the file.

Calculate the extended price (quantity \* price).

Keep a running total of extended prices and a count of the number of orders.

Calculate the average order amount (total extended price / number of orders).

Output: For each item, display the name, quantity, price, and extended price. Total of all extended prices. Total number of orders. Average order amount.

Problem 5

Input:

Text file with student last names, district codes (I or O), and the number of credits taken.

Process:

Read the student data from the file. Determine the cost per credit based on district code. Calculate the tuition owed (number of credits \* cost per credit).

Output:

For each student, display the last name, credits taken, and tuition owed. Sum of all tuition owed. Total number of students.