

1) Present Value

Suppose you want to deposit a certain amount of money into a savings account, and then leave it alone to draw interest for the next 10 years. At the end of 10 years you would like to have \$10,000 in the account. How much do you need to deposit today to make that happen? To find out you can use the following formula, which is known as the present value formula:

$$P = F / (1 + r)^n$$

The terms in the formula are as follows:

- P is the present value, or the amount that you need to deposit today.
- F is the future value that you want in the account (in this case, \$10,000).
- r is the annual interest rate (expressed in decimal form, such as .042).
- n is the number of years that you plan to let the money sit in the account.

Write a program with a function named `presentValue` that performs this calculation. The function should accept the future value, annual interest rate, and number of years as arguments. It should return the present value, which is the amount that you need to deposit today. Demonstrate the function in a program that lets the user experiment with different values for the formula's terms.

2) Stock Profit

The profit from the sale of a stock can be calculated as follows:

$$\text{Profit} = ((NS * SP) - SC) - ((NS * PP) + PC)$$

where NS is the number of shares, SP is the sale price per share, SC is the sale commission paid, PP is the purchase price per share, and PC is the purchase commission paid. If the calculation yields a positive value, then the sale of the stock resulted in a profit. If the calculation yields a negative number, then the sale resulted in a loss.

Write a function that accepts as arguments the number of shares, the purchase price per share, the purchase commission paid, the sale price per share, and the sale commission paid. The function should return the profit (or loss) from the sale of stock.

Demonstrate the function in a program that asks the user to enter the necessary data and displays the amount of the profit or loss.

3) Order Status

The Middletown Wholesale Copper Wire Company sells spools of copper wiring for \$100 each and ships them for \$10 apiece. Write a program that displays the status of an order. It should use two functions. The first function asks for the following data and stores the input values in reference parameters.

- The number of spools ordered.
- The number of spools in stock.

- Any special shipping and handling charges (above the regular \$10 rate).

The second function receives as arguments any values needed to compute and display the following information:

- The number of ordered spools ready to ship from current stock.
- The number of ordered spools on backorder (if the number ordered is greater than what is in stock).
- Total selling price of the portion ready to ship (the number of spools ready to ship times \$100).
- Total shipping and handling charges on the portion ready to ship.
- Total of the order ready to ship.

The shipping and handling parameter in the second function should have the default argument 10.00.

4) Overloaded Hospital

Write a program that computes and displays the charges for a patient's hospital stay. First, the program should ask if the patient was admitted as an in-patient or an out-patient. If the patient was an in-patient the following data should be entered:

- The number of days spent in the hospital
- The daily rate
- Charges for hospital services (lab tests, etc.)
- Hospital medication charges.

If the patient was an out-patient the following data should be entered:

- Charges for hospital services (lab tests, etc.)
- Hospital medication charges.

The program should use two overloaded functions to calculate the total charges. One of the functions should accept arguments for the in-patient data, while the other function accepts arguments for out-patient data. Both functions should return the total charges.

Input Validation: Do not accept negative numbers for any information.

5) Population

In a population, the birth rate is the percentage increase of the population due to births and the death rate is the percentage decrease of the population due to deaths. Write a program that asks for the following:

- The starting size of a population
- The annual birth rate
- The annual death rate
- The number of years to display

The program should then display the starting population and the projected population at the end of each year. It should use a function that calculates and returns the projected new size of the population after a year. The formula is

$$N = P(1 + B)(1 - D)$$

where N is the new population size, P is the previous population size, B is the birth rate, and D is the death rate.

Input Validation: Do not accept numbers less than 2 for the starting size. Do not accept negative numbers for birth rate or death rate. Do not accept numbers less than 1 for the number of years.

6) Kinetic Energy

In physics, an object that is in motion is said to have kinetic energy. The following formula can be used to determine a moving object's kinetic energy:

$$KE = 1/2 * mv^2$$

The variables in the formula are as follows: KE is the kinetic energy in joules, m is the object's mass in kilograms, and v is the object's velocity in meters per second.

Write a function named `kineticEnergy` that accepts an object's mass (in kilograms) and velocity (in meters per second) as arguments. The function should return the amount of kinetic energy that the object has. Demonstrate the function by calling it in a program that asks the user to enter values for mass and velocity.

7) Winning Division

Write a program that determines which of a company's four divisions (Northeast, Southeast, Northwest, and Southwest) had the greatest sales for a quarter. It should include the following two functions, which are called by main.

- double **getSales()** is passed the name of a division. It asks the user for a division's quarterly sales figure, validates the input, then returns it. It should be called once for each division.
- void **findHighest()** is passed the four sales totals. It determines which is the largest and prints the name of the high grossing division, along with its sales figure.

Input Validation: Do not accept dollar amounts less than \$0.00.

8) Retail Price

Write a program that asks the user to enter item's wholesale cost and its markup percentage. It should then display the item's retail price. For example:

- If an item's wholesale cost is 5.00 and its markup percentage is 100%. Then the item's retail price is 10.00.
- If an item's wholesale cost is 5.00 and its markup percentage is 50%. Then the item's retail price is 7.50.

The program should have a function named **calculateRetail** that receives the wholesale cost and the markup percentage as arguments, and returns the retail price of the item.

Input Validation: Do not accept negative values of either the wholesale cost of the item or the percent markup.

9) isPrime Function

A prime number is an integer greater than 1 that is evenly divisible by only 1 and itself. For example, the number 5 is prime because it can only be evenly divided by 1 and 5. The number 6, however, is not prime because it can be divided by 1, 2, 3, and 6.

Write a Boolean function named **isPrime**, which takes an integer as an argument and returns true if the argument is a prime number, and false otherwise. Demonstrate the function in a complete program.

- * TIP: Recall that the % operator divides one number by another, and returns the *
- * remainder of the division. In an expression such as num1 % num2, the % operator *
- * will return 0 if num1 is evenly divisible by num2

10) Star Search

A particular talent competition has 5 judges, each of whom awards a score between 0 and 10 to each performer. Fractional scores, such as 8.3, are allowed. A performer's final score is determined by dropping the highest and lowest score received, then averaging the 3 remaining scores. Write a program that uses these rules to calculate and display a contestant's score. It should include the following functions:

- void **getJudgeData()** should ask the user for a judge's score, store it in a reference parameter variable, and validate it. This function should be called by main once for each of the 5 judges.
- double **calcScore()** should calculate and return the average of the 3 scores that remain after dropping the highest and lowest scores the performer received. This function should be called just once by main, and should be passed the 5 scores.

The last two functions, described below, should be called by **calcScore**, which uses the returned information to determine which of the scores to drop.

- int **findLowest()** should find and return the lowest of the 5 scores passed to it.
- int **findHighest()** should find and return the highest of the 5 scores passed to it.

Input Validation: Do not accept judge scores lower than 0 or higher than 10.

11) Lowest Score Drop

- Write a program that calculates the average of a group of test scores, where the lowest score in the group is dropped. It should use the following functions:
- void **getScore()** should ask the user for a test score, store it in a reference parameter variable, and validate it. This function should be called by main once for each of the five scores to be entered.
- void **calcAverage()** should calculate and display the average of the four highest scores. This function should be called just once by main, and should be passed the five scores.
- int **findLowest()** should find and return the lowest of the five scores passed to it. It should be called by calcAverage, which uses the function to determine one of the five scores to drop.

Input Validation: Do not accept test scores lower than 0 or higher than 100.

12) Safest Driving Area

Write a program that determines which of 5 geographic regions within a major city (north, south, east, west, and central) had the fewest reported traffic accidents last year. It should have the following two functions, which are called by main.

- int **getNumAccidents()** is passed the name of a region. It asks the user for the number of traffic accidents reported in that region during the last year, validates the input, then returns it. It should be called once for each city region.
- void **findLowest()** is passed the five accident totals. It determines which is the smallest and prints the name of the region, along with its accident figure.

Input Validation: Do not accept an accident number that is less than 0.