Programming Fundamentals
Practice: Conditional Structure

PRG-1. Write a program that accept only two-digit unsigned integers from the user and then reverses the digits and prints them. Use an *if* statement to terminate the program if the input number has more than two digits.
PRG-2. Write a program that, given three integers, prints the smallest one.
PRG-3. Write a program that, given an integer between 1 and 12 (inclusive), prints the corresponding month of the year.
PRG-4. Write a program that, given the type of vehicle ('c' for car, 'b' for bus, 't' for truck) and the hours a vehicle spent in the parking lot, returns the parking charge based on the rates shown below.

car: $2 per hour bus: $3 per hour truck: $4 per hour

PRG-5. Write a program that determines a student's grade. It reads three test scores (Between 0 and 100) and calculates the grade based on the following rules:

a) If the average score is 90 or more, the grade is 'A'.
b) If the average score is between 80 and 90, the program checks the third score.
c) If the third score is more than 90, the grade is 'A'; otherwise, the grade is 'B'.
d) If the average score is between 70 and 80, the program checks the third score.
e) If the third score is more than 80, the grade is 'B'; otherwise, the grade is 'C'.
f) If the average score is between 60 and 70, the program checks the third score.
g) If the third score is more than 70, the grade is 'C'; otherwise, the grade is 'D'.
h) If the average score is less than 60, the program checks the third score. If the
i) third score is more than 60, the grade is 'D'; otherwise, the grade is 'F'.

PRG-6. Write a program that calculates and prints a student's total tuition at a college. The students pay a fee of $10 per unit for up to 12 units; once they have paid for 12 units, they have no additional per-unit fee. The registration fee is $10 per student.
PRG-7. A wholesale store gives a discount on the number of items purchased, as shown below.

Quantity Discount Quantity Discount
1 to 9 0% 50 to 99 5%
10 to 49 3% 100 or more 10%

Write a program that, given the quantity and unit price of an item, calculates the total price after the discount.
PRG-8. Write a program that prints the quarter (1, 2, 3, and 4) of a point in the Cartesian (rectangular) system given the values of *x* and *y* for the point. For example, if both *x* and *y* are positive, the point is located in the first quadrant. If both *x* and *y* are negative, the point is located in the third quadrant, and so on.
PRG-9. Change Program 4.17 to consider the extra day in a leap year (February is 29 days instead of 28 days). The program must get the year from the user and use the following formula to find if the year is a leap year.

leap Year = (year % 400) || (year % 4 &&! (year % 100))

PRG-10. Write a program that finds the day of the week for any given date using Zeller's congruence. Zeller found the following formula to calculate the day of the week using the day of the month, the month, and the year.

weekday = (day + 26 * (month + 1) / 10 +
year + year / 4 − year /100 + year /400) % 7

The formula is based on the following:

a) There are seven days in the week so the calculation must be done modulo 7.
b) The first term, *day*, shows that each day of the month moves the weekday
c) forward by 1.
d) The second term, 26 * (*month* + 1) / 10, is the Zeller congruence. Instead of worrying about the number of days in each month, Zeller devised this solution. Zeller's formula works for a solar system when the year starts from March, not January. Therefore, we must consider the months January and
e) February as months 13 and 14 of the previous year. In other words, January of 2017 must be considered as month 13 of *year* 2016.
f) The next term, *year*, moves the week day one day for each year because **a non-leap year** is 365 **and** when divided by 7 the result is 1.

g) The next term, *year* / 4, is the contribution of the leap year. We know that each year divided by 4 **could** be a leap year **that adds** one day to the week day.
h) The next term, *year / 100,* is the term that excludes the year divided by 100 from the previous year.
i) The next term, *year / 400,* is the term that defines if a year divided by 400 is a leap year and needs to add 1 to the calculation.

PRG-11. Change the menu-driven calculator program (Program 4.18) to do calculations on floating-point numbers.

PRG-12. Write a program that, given a dollar value, prints the minimum number of bill denominations in 100s, 50s, 20s, 10s, 5s, and 1s. Use the conditional expression (:) to print only nonzero numbers of bills.