Unit 3: Conditional Statements | **Application: The Zodiac**

In this application for Unit 3 you will be developing a program that can tell the user in what sign of the zodiac an entered date lies.

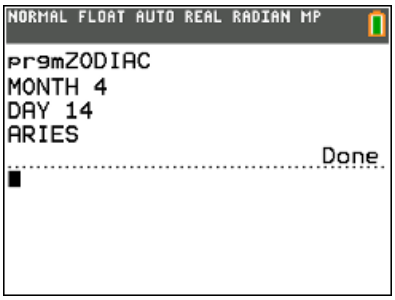
Objectives:

- Work with date conversions.
- Use **If** statements to determine the sign of the zodiac for a given date.
- Work with **String** variables.

The Zodiac

In astronomy and astrology the **zodiac** is a division of the sky into 12 equal regions. The regions are named by the constellations that are approximately within these regions. The Babylonians developed this division around 1,000–500 B.C. They began their calendar year with the vernal equinox (the first day of Spring), hence the first sign of the zodiac is Aries and covers the period from March 21 to April 20.

The Program



In this lesson we'll write a program that lets a user enter a month and a day and then the program displays the sign of the zodiac for that date. This program utilizes many **If…Then** statements.

The user will enter the month number and day number according to our calendar, and the program will convert the date into the Babylonian format (March = 1, April = 2, … , January = 11, February = 12). The tests for each sign can be complex. For example:

If (M=1 and D>20) or (M=2 and D<21)

To simplify the programming we'll develop a single numeric value (a *code*) to represent the dates. This code makes it easier to write the **If** statements rather than have to deal with more complex expressions involving **and**s and **or**s.

Pseudocode

When developing a large program it is often helpful to start with a plain-English outline of the program. This is called 'pseudocode' because it is not written using a specific programming language. The outline is written in a way that makes it easy to convert to a programming language later on.

Here's an outline (pseudocode) of the zodiac program:

PROGRAM: ZODIAC
Enter the Month user enters a number from 1 to 12
Enter the Day user enters a number from 1 to 31

(The zodiac signs begin with Aries so we make March the first month and January and February are months 11 and 12.)
Subtract 2 from the Month.
If the Month is less than 1 then add 12 to the Month.

Create a one-number Code for the date (combine Month and Day into one number, the *Code*, whose first one or two

digits are the month and whose last two digits are the day). We accomplish this by multiplying the month by 100 and add the day. For example: July 4 is Babylonian Month 5, Day 4 so the Code is 100*5+4 = 504.

Multiply the Month by 100 and add the Day and then store the result in the Code variable.

Store "Invalid" in a **String*** variable. This variable will be used later to display the sign or the word *Invalid*. See ***Strings** below.

Now check to see in which zodiac sign the Code belongs:

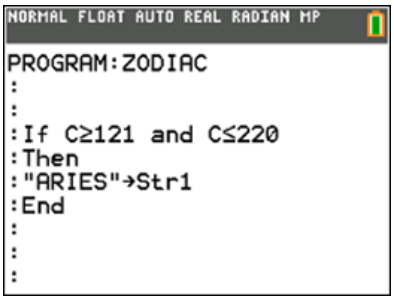
if Code >=121 and Code<=220 *this represents the days from March 21 to April 20*

then

store "Aries" in the string variable **Note the quotation marks!**

end

Write one of these structures for each of the twelve signs of the zodiac.



After the 12 **If** structures the string variable will contain either "INVALID" or one of the signs, so…

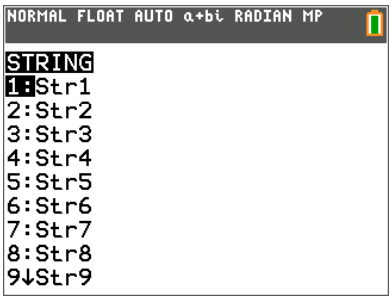
Display the string variable.

Here are the dates for each sign:

Aries:	March 21–April 20
Taurus:	April 21–May 21
Gemini:	May 22–June 21
Cancer:	June 22—July 22
Leo:	July 23—August 22
Virgo:	August 23—September 23
Libra:	September 24—October 23
Scorpio:	October 24—November 22
Sagittarius:	November 23—December 21
Capricorn:	December 22—January 20
Aquarius:	January 21—February 19
Pisces:	February 20—March 20

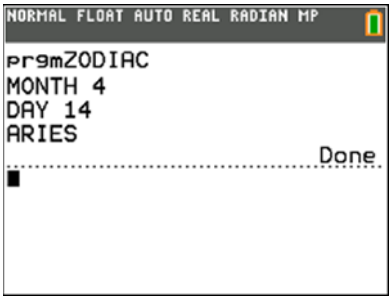
***Strings**

This program stores a *string of characters* such as "INVALID" in a **String** variable. The TI-84 has 10 string variables for you to use. To access these variable names press the [VARS] key and select the **String…** menu. Be sure to use only one string variable for all the signs of the zodiac.



Your Task

Write the program **ZODIAC** that meets the above specifications. The program should produce something like what is displayed here.



Caution: *Pisces* takes a special condition!

Extension:

The program does not check to see if the month and day entered are legal dates. Add **If** statements after the input section to make sure that the values entered are 'legal'.

Tip: Remember, some months have 30 days, some have 31, and one has only 29. Also, there is no month 13 or 14. See what happens if you enter 14 for the month.