

Logo Activity 11

More Decisions

You have used the IF command to write procedures that branch. You use IF when you want to make a decision within a procedure. Two things can happen when you use an IF command. Either the condition after the IF is true or it is false. If the condition is true, the list of commands following the IF command is run. If the condition is false, Logo moves to the next instruction in the procedure.

Creating a Case statement for a MENU

Sometimes you want a procedure to act like a menu and, in addition, to do nothing at all if an incorrect choice is made. Type in the following procedures:

```
TO MENU
  SHOW.CHOICES
  WHICH.ONE READCHAR
  WAIT 50
  MENU
END

TO SHOW.CHOICES
  CS
  CT
  PRINT [1. SEE A TRIANGLE]
  PRINT [2. SEE A SQUARE]
  PRINT [3. SEE A PENTAGON]
END

TO WHICH.ONE :CHOICE
  IF :CHOICE = 1 [TRIANGLE STOP]
  IF :CHOICE = 2 [SQUARE STOP]
  IF :CHOICE = 3 [PENTAGON STOP]
  PRINT [BAD INPUT, TRY AGAIN]
END

TO TRIANGLE
  REPEAT 3 [FD 30 RT 120]
END

TO SQUARE
  REPEAT 4 [FD 30 RT 90]
END

TO PENTAGON
  REPEAT 5 [FD 30 RT 72]
END
```

Type Menu to test the set of procedures. Here how it works:

1. Menu calls SHOW.CHOICES
2. SHOW.CHOICES displays the three menu items
3. The call to WHICH.ONE reads in a character (1,2,or 3) and passes it to CHOICE in WHICH.ONE
4. IF CHOICE (the key pressed) is a 1, TRIANGLE is called (draws a triangle) followed by a STOP. STOP ends the WHICH.ONE procedure and returns processing to MENU. Menu's next command is a WAIT 50 followed by a recursive call to itself (MENU).
5. Same as 4, except SQUARE is called.
6. Same as 4, except PENTAGON is called.

Two Action IFELSE Statement

You may wished that you could write a procedure that causes Logo to take one action if the condition after the IF is true and another if the condition is false. The following statement will help you understand the IFELSE statement.

```
TO EXPERIMENT
  IFELSE ((RANDOM 2) = READCHAR ) [PRINT [LUCKY GUESS]] [PRINT [NOT A CHANCE!]]
END
```

Type EXPERIMENT to test the above procedure and enter 1 to see if you guessed the randomly generated number (0 or 1).

Password Checking

The procedure PASSWORD below uses an IFELSE with two possible actions. Test the procedure by typing a bogus password and then try it with the correct password (SECRET). The IFELSE must appear on one line.

```
TO PASSWORD
  CS CT
  TYPE [PLEASE TYPE THE PASSWORD TO CONTINUE -->]
  IFELSE (READLIST = [SECRET]) [PRINT [YOU MAY CONTINUE]
    DESIGN] [PRINT [WRONG PASSWORD] WAIT 10 PASSWORD]
END
```

```
TO DESIGN
  REPEAT 4 [FD 40 RT 90]
  IFELSE HEADING = 350 [STOP] [RT 10 DESIGN]
END
```

If the user types SECRET for the READLIST the first list of instructions will be executed. This includes two statements. One placing a message on the user screen and the other executing a procedure called DESIGN. If a user types something other than SECRET the second list of instructions will be executed. The user will be told that the PASSWORD is incorrect and will be allowed to try again!

The second procedure also contains an IFELSE. In this procedure we see a new reporter called HEADING. It reports the turtle's present direction! If the HEADING equals 350, the procedure STOPS; otherwise a RT 10 is executed and DESIGN is executed again.

Calling Procedures With IFELSE

It is acceptable to put a few statements in each list after an IF or IFELSE. However, if those lists get very long, it gets quite difficult to read. It is better to write a separate procedure for each list. For example, rather than:

```
IFELSE (READWORD = 1) [FORWARD 50 RT 90] [BACK 50 LEFT 90]
```

it might better read:

```
IFELSE (READWORD = 1) [GO.FD] [GO.BK]
```

```
TO GO.FD  
  FD 50 RT 90  
END
```

```
TO GO.BK  
  BK 50 RT 90  
END
```

Assignment # 11

Name _____

1. Create a menu that will allow a user to pick one of four menu choices and then perform the desired command. The choices are as follows:
 - a) Clear the graphic screen.
 - b) Move Forward 100.
 - c) Turn Right 90.
 - d) Turn Left 90.

If an incorrect value is entered the user should be told that a bad value was entered. After each entry clear the text screen and present the menu choices again. Record your procedures here!

2. Write a procedure to GUESS a two digit number between 0 and 99. Remember to load READWORD. Report your procedure here.

3. Create and record a procedure called TEST that has three IF statements that determines whether a number is greater than 10, less than 10, or equal to 10. If a user typed:

TEST 16

The output on the page would say:

This number is greater than 10.

Extra Credit (2 pts)

1. In [assignment 9](#) you created a procedure to keep saying “Keep your hands off the keys!” appearing on the screen. [Click on this link to see how you will need to modify the program and make it behave.](#)

You will need to understand a couple of additional commands for this problem.

Hints:

```
TO EXPERIMENT
  SETFOCUS [MSWLOGO SCREEN] ← sets the focus to the graphics window
  KEYBOARDON [PRINT (CHAR KEYBOARDVALUE) = "Y ] ← Outputs True or False
END
```

As you can see, pressing the “y” produces a True and pressing any other key produces a False.

You may use and/or modify the following code for this problem.

```
TO MESSAGE
  SETFOCUS [MSWLOGO SCREEN]
  KEYBOARDON [IFELSE ((CHAR KEYBOARDVALUE) = "Y ) [HALT] [MESSAGE] ]
END
```

If someone presses “y”, the HALT command stops the processing of the program.

Record the procedure(s) for your solution below.