Skill Builder 2: Turning

Overview:

Students will write a program on their calculator to turn the Rover left and right at various angles. They are challenged to make their Rover slowly turn like the hands of clock and to report the time of day on their calculator.

Goals:

Students will:

1. write a TI Python program to turn Rover left or right at different angles.
2. incorporate the For loop control structure into a program.
3. incorporate the print() function into a program.
4. make a model of a clock using the Rover.

Background:

The Rover turns by rotating its wheels in opposite directions at the same speed at the same time. This type of turn is called a spin because it spins in a circle that has a center at the midpoint of the two wheels. This midpoint is also the location of the marker tip when a dry-erase marker is inserted into the pen holder. When the Rover performs a turn, the program needs to inform the motors the direction and size of the spin. The direction of the turn is determined by using the rv.left() and rv.right() functions from Rover Drive menu. The direction is from the view of as if Rover had a driver's seat. The size of the turn is determined by the angle, this is a value in degrees, radians, or gradians. A full spin is 360 degrees. This number is from the base 60 sexagecimal system used by the Sumerians in ancient Babylon. Similarly, a full spin is 2π radians. This number comes from the fact that the angular width of an arc of one radius in length along the circumference of any circle is defined as one radian. Also, a full spin is 400 gradians. The gradian is defined in the metric system as 1/100 of a circle quadrant. The default angle measurement unit is degrees. The Rover can accept all three units when using rv.left(angle,"unit") and rv.right(angle,"unit") from the Rover Drive with Options menu.

Rover Function	Example	Behavior
import module_name as name_space	`import ti_rover as rv`	Required for all TI Rover Python programs. Imports the ti_rover module into the Python program. The module provides the methods for controlling the Rover. The import ti_rover statement is available from the Rover menu.
from module_name import *	`from time import *`	Imports all the functions in the time module for use in the program. The time module includes the sleep() method. The import time statement is available from the More Modules Time menu.
rv.right (angle_degrees) rv.left (angle_degrees)	**rv.right**() **rv.right**(45)	Rover spins to the right 90 degrees (If no value is entered for angle_degrees, rv.right() and rv.left() use the default value of 90), followed by a spin of another 45 degrees to the right. To input rv.right without and angle value: choose rv.right(angle) from the Rover Drive menu, then press delete, then right arrow, then enter to move to the next line. For angle units other than degrees, use rv.right(angle,"unit") from the Drive with Options menu. The choices for angle units are degrees, radians and gradians.

print(value or "text string")	n=3 print(n) print("number= ") print("number= ",n)	Prints value of variable n, which is 3; then on the next line prints the text string "number= "; finally prints "number= " followed by the value of variable n all on the next line.
sleep(seconds)	**sleep(1.5)**	Pauses program for 1.5 second. The sleep() function can be found on the Rover Commands menu.
for index in range(stop value): block	for n in range(10): print(n)	Repeats the statements in the block ten times, printing the value of the index variable, n, as 0,1,2,...9. The index variable, n, starts at 0 and increases by 1 with each loop. If n is less than the stop value, 10, the loop continues to repeat. The block starts with a colon and includes the indented lines that follow. The for loop statement is found on the Built-ins Control menu.

* The LEFT and RIGHT turns are made with a frame of reference from Rover's driver's seat.

** Radians is an angular unit of measure used in mathematics. There are 2π RADIANS in 360° DEGREES.

*** Gradians is an angular unit of measure also used in mathematics. There are 100 GRADIANS in a quarter circle; hence 400 grads in a full circle.

Setup Rover:

Students may work in groups of two or three. Choose an area to work that has at least 2 meters of clear uniform floor space. Carpeted flooring is less desirable than tile. If needed, driving mats may be used as a driving surface.

Supplies:

- Masking tape
- Drive mats

Student Activity

Challenge 1: Write a program named "**c1**" that uses right turns to spin Rover a total of 360 degrees.

Teacher Activity

Guidance during challenge 1:

Review the TI Python editor and how to run a program with students.

- Basic navigation on the calculator.
- Saving and opening files.
- Editing new and existing programs.
- Running programs.
- Editing program features.
- Do the first challenge just after introducing the rv.right() and rv.left() functions. Do not yet, inform the students about the option to enter a value for the turn angle.
  - Note: To input rv.right() and rv.left() without an angle value: paste the functions from the menu, then with the "angle" prompt inside the parentheses press the delete key. Use arrow keys to move to the end of the line and press enter to move to the following line.
- As the students explore using the rv.right() and rv.left() functions without inputting an angle value,

challenge them to determine what angle the Rover is turning. Next, ask how many angle turns are needed to spin a circle?

- Discuss and challenge students that are ahead how they could do the same program turning to the left.
- See program c1 in On-Ramp to Robotics Unit 1 Skill Builder 2 CXII Python.tns.

Challenge 2: Write a program named "c2" to turn Rover in a circle using the rv.right() or rv.left() functions. After each turn, display the total angle turned from the starting point on the calculator screen.

Guidance during challenge 2:

Review the usage of the print() function on the Built-ins I/O (Inputs/Outputs).

- Menu->4: Built-ins ->6: I/O ->1: print()
- Enter the display string in quotes " "
- An example: print("Turn Angle is 90")

Review the usage of the sleep() function.

- Menu-> 9: Rover-> 7: Commands->1: sleep()
- Enter the time in seconds to wait.
- An example: sleep(2)
- The Rover and the calculator perform at different speeds. When a drive command is sent to the Rover, it may require several seconds for the Rover to drive that command. The program must wait for the Rover to finish before doing another command. The Wait command is used to tell the calculator to do nothing and just wait for the Rover to finish driving.
- Have the students explore the effect of using the sleep function in different places in their program and also with different times. It can be tricky to keep the print() in sync with the Rover's motion in order to achieve Challenge 2. Please refer to program c2 On-Ramp to Robotics Unit 1 Skill Builder 2 CXII Python.tns for example solution for this challenge.
- Discuss and challenge students who are ahead, how they could do the same program in radians or gradians?

Challenge 3: Write a program named "**c3"** and use a For loop to turn three circles to the right and then three circles to the left in steps of 90 degrees. Display the total of degrees turned by the Rover at each step on the calculator screen.

Guidance during challenge 3:

Review the usage of the for index in range() control function.

- Menu->4: Built-ins ->2: Control ->1: for index in range()
- The for loop control structure enables the program a repeat a set of statements a fixed number of times.
- The for loop is defined by four values: an index variable to count the number of iterations, a start value for the index variable, an end value for the index variable and a step value for the index variable. The for index in range() statement assumes a start value of 0 and an index step value of 1. In Python, the

statements to be repeated are defined by a beginning colon and statements that are indented from the row of the statement setting up the for control structure.

```
for n in range(5):
  set of statements to repeat (note: indented from the for loop statement)
statement that is not part of the loop (note: at same indent level as for loop statement)
```

- Help the students to understand the difference between using the previous program which explicitly calls out each command and the use of the for loop which reuses one set of statements repeatedly. Also, inform students of the use of the index variable. This variable may be used to calculate the total angle turned by the Rover using the expression (n+1)*90, where n is incremented each time the for loop is iterated. See example program c3 in On-Ramp to Robotics Unit 1 Skill Builder 2 CXII Python.tns.

Challenge 4: Write a program named "**c4"** that has Rover model the hour hand on a clock. Turn the Rover to stop at each hour of the clock. Display the value of the current hour on the calculator display.

Guidance during challenge 4:

- This challenge is the final challenge of this activity and requires that students incorporate all of the skills learned so far. Encourage student to look back through programs c1, c2, and c3 as references for creating this new program. Students may be curious of the fact that there are 24 hours in a day. This number, like 360 degrees in a circle, comes from the Babylonian sexagecimal system since 360/15=24. Se example program c4 in On-Ramp to Robotics Unit 1 Skill Builder 2 CXII Python.tns