Teacher Instructions

The idea here is that you print a few sets of these cards and let your students choose which programs to try. This allows for differentiated instruction. You could also have the student rotate through and complete all tasks if desired.

Task Cards

Functions

Task 1

For exercise to be beneficial, your exercising heart rate must be in your training heart rate zone. To calculate your training zone you will need to find your maximum heart rate (MHR) and then do some additional computations to get your zone.

Write a program that asks for the user's age and then compute their THR zone. Your program should calculate and print out the THR zone as well as the MHR.

Use a function to calculate a THR zone minimum value and another to compute the maximum values:

- Calculate maximum heart rate (MHR) as 220 – age
- Minimum training zone value is approximately 60% of MHR
- Maximum training zone value is approximately 75% of MHR.

The majority of your workout should be within this zone, and your maximum heart rate is MHR.

Task 2

Write a program to play a game. If the user answers a simple mathematics question correctly a function called '**Correct'** is called and if the questions is not correct a function called, '**Incorrect'** is called**.** Your functions do not have any arguments and therefore do not need to return anything. You decide what happens inside each function.

Task 3

Write a program that allows the user to enter an amount of money in change. Your program should print out the minimum number of coins in this amount of money. Your program should include 3 functions called **quarters, dimes,** and **nickels**. If 128 is entered the output should be 128 cents is:

4 quarters
2 dimes
1 nickels
3 pennies

Task 4

Write a program that makes Rover drive away for a few seconds, stop, change LED color, drive back to the user, stop, change LED color, and make a sound to indicate mission complete. Create a user-defined function called Move() that is called by the main program to make the Rover actually complete the tasks required

Task 5

Write a program that makes Rover trace out an equilateral bowtie. Then, extend your program so that Rover traces out the original equilateral bowtie and a scaled version of it.

You should create functions for each of the tasks asked for above and have your main program call those functions to complete the tasks.

Task 6

Draw a path or maze for Rover to navigate, then write a program that moves Rover along the path or through the maze.

Create a user-defined function that will allow the Rover to navigate your path.

Task 7

Write a program that makes Rover operate like a vehicle that beeps when it is driving in reverse, but not when it is driving forward. Extend your program to “randomly” drive forward or backward.

Create user-defined functions to complete each task.

Task 8

Write a program that makes Rover map out a specific path of your choosing/design.
Create a function to perform this task.

Task 9

Write a program that makes Turtle trace out an equilateral bowtie. Then, extend your program so that Turtle traces out the original equilateral bowtie and a scaled version of it.
Create user-defined functions for each of the tasks asked for above and have your main program call those functions to complete the tasks.

Task 10

Draw a path or maze for Turtle to navigate, then write a program that moves the Turtle along the path or through the maze.

Create a user-defined function that will allow the Turtle to navigate your path.

Task 11

Write a program that makes Turtle map out a specific path of your choosing/design.
Create a function to perform this task.