PRIME LESSONS

By the Makers of EV3Lessons

RECURSION

BY SANJAY AND ARVIND SESHAN

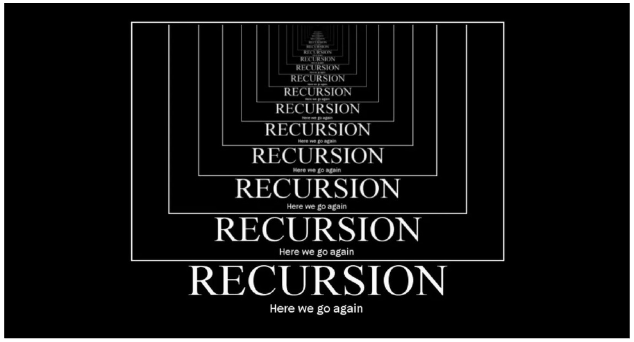
This lesson uses SPIKE 3 software

LESSON OBJECTIVES

- Learn how to create recursive functions

Copyright © 2021 Prime Lessons (primelessons.org) CC-BY-NC-SA. (Last edit: 09/09/2023)

INTRO TO RECURSION



- Definition recursion (n):
  - see recursion
- The definition refers to itself (like a loop)
- Some famous examples are:
  - Fibonacci series: $f_n = f_{n-1} + f_{n-2}$ *1, 1, 2, 3, 5, 8, 13, …..*
  - Factorial: $n! = n * (n-1)!$ *5! = 5*(4*(3*(2*(1)))) = 120*
- In Python: a function that calls itself

Copyright © 2021 Prime Lessons (primelessons.org) CC-BY-NC-SA. (Last edit: 09/09/2023)

PROGRAMMING A RECURSIVE FUNCTION

- There are two parts to recursion:
    - The base case → a known case
        - Sometimes there are multiple base cases
    - The recursive case → everything else

```
def recursiveFunction():
    if (this is the base case):
        return something non-recursive
    else:
        return something recursive
```

Copyright © 2021 Prime Lessons (primelessons.org) CC-BY-NC-SA. (Last edit: 09/09/2023)

RECURSION: FACTORIAL

- Base Case: factorial(1) = 1 (i.e. 1! = 1)
- Recursive case: return n*(factorial(n-1))

```
def factorial(n):
   if (n == 1):
       return 1
   else:
      return n*factorial(n-1)
```

Copyright © 2021 Prime Lessons (primelessons.org) CC-BY-NC-SA. (Last edit: 09/09/2023)

RECURSION: FIBONACCI

- Base Case 1: fibonacci(1) = 1
- Base Case 2: fibonacci(2) = 1
- Recursive case: return fibonacci(n-1)+fibonacci(n-2)

```
def fibonacci(n):
   if (n == 1):
       return 1
   elif (n == 2):
       return 1
   else:
      return fibonacci(n-1) + fibonacci(n-2)
```

Copyright © 2021 Prime Lessons (primelessons.org) CC-BY-NC-SA. (Last edit: 09/09/2023)

CHALLENGE: PELL SEQUENCE

- Create a recursive function to get the nth value in the Pell sequence
- The Pell sequence is 0, 1, 2, 5, 12, 29, 70, 169, 408, 985, ……
- Mathematically, it is defined as

$$P_n = 2 * P_{n-1} + P_{n-2}$$

- Print the 5th PELL number to the light matrix

Copyright © 2021 Prime Lessons (primelessons.org) CC-BY-NC-SA. (Last edit: 09/09/2023)

CHALLENGE SOLUTION

```
from hub import light_matrix

import runloop, sys


# Function to stop the program using a system exception

def stopAndExitProgram():

    sys.exit("Stopping")


def PELL(n):

    if (n < 1):

        return "Invalid"

    elif (n <= 2):

        return n - 1

    else:

        return 2 * PELL(n-1) + PELL(n-2)


async def main():

    await light_matrix.write(str(PELL(5))) # convert number to str before writing

    stopAndExitProgram()


runloop.run(main()))
```

Copyright © 2021 Prime Lessons (primelessons.org) CC-BY-NC-SA. (Last edit: 09/09/2023)

CREDITS

- This lesson was created by Sanjay and Arvind Seshan for Prime Lessons
- Additional contributions by FLL Share & Learn community members
- More lessons are available at www.primelessons.org

This work is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License.