Unit 5 Procedures & Functions

## Learning goals

1. Describe the differences between procedures and functions
   1. Procedures change the state of the program but return nothing, while functions return a value but do not change the state of the program.
   2. Procedure-calls stand in isolation, whereas function-calls are either on the RS of an assignment statement or passed as arguments to other procedures and functions.
2. Define functions and procedures that take numbers, booleans and strings as parameters, as well as arrays thereof.
3. Define functions that return numbers, booleans, strings and graphical objects.
4. Define procedures that print to the screen and produce graphics and animations.
5. Define procedures and functions that call other procedures and functions.
6. Distinguish between global and local variables and use them correctly in programs.
7. Trace a chain of nested procedure- & function-calls.
8. Use functions and procedures to solve applied problems, e.g. calculate the number of days between two given dates.
9. Use stepwise refinement to break a program into a short sequence of simpler, high-level steps.

|  |  |
| --- | --- |
| **Lesson** | **Content** |
| **Lesson 1** Intro to functions | - Python functions and procedures you know - Differences between functions and procedures - Defining a getAverage function - Anatomy of a function: parameters, body, return-statement - Function calls: arguments and placement within a program - Python demo: getArea, getAverageOfArray - Practice: Function Exercises.doc |

|  |  |
| --- | --- |
| **Lesson 2** Intro to procedures | - Recap of functions: getDistance(x1, y1, x2, y2) - Defining a drawCircle procedure - Procedure calls: placement within a program - Python demo: Draw Circle.py and Cloud Drawing.py  - Practice: Procedure Exercises 1 and 2.doc |
| **Lesson 3** Nested procedures and functions | - Simple examples of tracing - Solar system animation.py and how it uses multiple procedures - Finish Procedure Exercises 2.doc |
| **Lesson 4** Stepwise refinement 1 | - Intro the Cheque Writer problem - Why this is a hard problem (16 vs. 26 vs. 36, and the zero skipping) - Manipulating strings in Python using *String Tools.py* - Design threeDigitSpeller() using oneDigitSpeller() & twoDigitSpeller() - Begin using *Cheque Writer GETTING STARTED.py* as a starting point. |
| **Lesson 5** Global and local variables | - Metaphor: Personal vs. private information - Definition of global and local variables - Examples - Finish Cheque Writer problem |
| **Lesson 6** Intro to 1st team project | - Take up solution to Cheque Writer problem - Introduce the Sink The Penguins problem - Demonstrate *Sink The Penguins GETTING STARTED.py* - Assign teams of two - Announce quiz |
| **Lesson 7** Team project work day | - Quiz on functions and procedures - Work on Sink The Penguins |
| **Lesson 8** Team project work day | - Oscar awards for best Assignment 4 animations - How to animate an explosion - Work on Sink The Penguins |
| **Lesson 9** Team project work day  (if needed) | - Finish Sink The Penguins |