|  |  |
| --- | --- |
| **Programming 1 (PRG1)**  Diploma in IT / DS / CSF / IM / CICTP  Year 1 (2023/24) Semester 1 | Week 12 |
|  |
| **Exercise 12 : Functions** | |

**OBJECTIVES**

At the end of this exercise, students should be able to develop Python programs with:

* Functions

**IMPORTANT**

* Create a folder, **Week12**, in your hard disk.
* For programming questions, create Python programs with the given file names in the **Week12** folder created above. Do add the description, your name and student ID as comments at the beginning of each program.
* For non-programming questions, type your answers in the boxes provided below the questions.
* At the end of the session, compress all the files in your **Week12** folder (i.e. the Python program files and this word document) and submit the zip file in POLITEMall.

**Activity 1**

Temperature Conversion - ( file name: convert\_temperature.py )

* Write a function, convert\_temperature(), to convert a temperature from Celsius to Fahrenheit.

The formula is given below:

f = (c \* 9 / 5) + 32

* Write code to use and test the function.
* Sample run:



**Activity 2**

Extreme Values - ( file name: get\_extremes.py )

* Write a function get\_extremes() that receives a list of integers as a parameter, and returns the smallest and largest values in the list.

Example function call:

num\_list = [ 10, -13, 50, 5, 7, 65, -40, 44, 30 ]

smallest, largest = get\_extremes(num\_list)

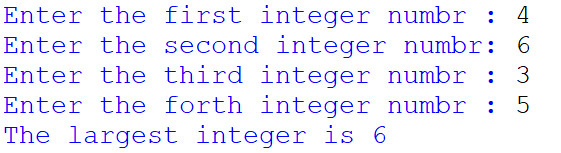
Hint: you can import the math module and use math.inf for infinity, and –math.inf for negative infinity

* Write code to test the function.

**Activity 3**

Find Largest Value - ( file name: find\_largest.py )

* Write a function find\_larger(n1, n2) that takes in two integer parameters, and returns the larger of the two.
* Then write some code to call find\_larger(), to find the largest integer among four integers given by user.
* Sample run:



**Activity 4**

Even Integers – ( file name: even\_integers.py )

* Write a function is\_even(n) that receives an integer parameter, checks if it is even and returns true if even, false if not.
* You are given a list of 10 integers. You are required to write code to make use of is\_even() to check if the integers are even, and display those that are so.

num\_list = [ 10, -13, 50, 5, 7, 24, 65, -40, 44, 30 ]

**Activity 5**

Even Integers 2 – ( file name: even\_integers2.py )

* With respect to question 4, consider the case that find\_even() is now able to print out the integer if it is even. Modify the code accordingly to print the even integers in the list.

**Activity 6**

Power – ( file name: power.py )

* Write a function power(x, n) that receives a non-zero integer parameter and another integer parameter, representing base and exponent respectively, and returns the value of xn.
* For example, power(3, 4) will calculate 34 or (3 \* 3 \* 3 \* 3).
* Write code that reads in base and exponent from user, finds out and displays the result making use of function power().

**Activity 7**

Square of Asterisks – ( file name: square\_of\_asterisks.py )

* Write a function print\_square(side)to display a solid square of asterisks (same number of rows and columns) whose side is specified in integer parameter, side.

You may make use of the print\_character() function in lecture 12-1 slide 8

For example, if side is 4, the function should display

\* \* \* \*

\* \* \* \*

\* \* \* \*

\* \* \* \*

* Write code to read in side from user, and make use of the function to output the pattern.

**Activity 8**

Square of Characters – ( file name: square\_of\_characters.py )

* Modify the function in the question 7 to display the square with any given character, char, included in the parameter as shown in the function header below.

print\_square(side, char)

* Thus if side is 5 and character used is ‘#’, then

# # # # #

# # # # #

# # # # #

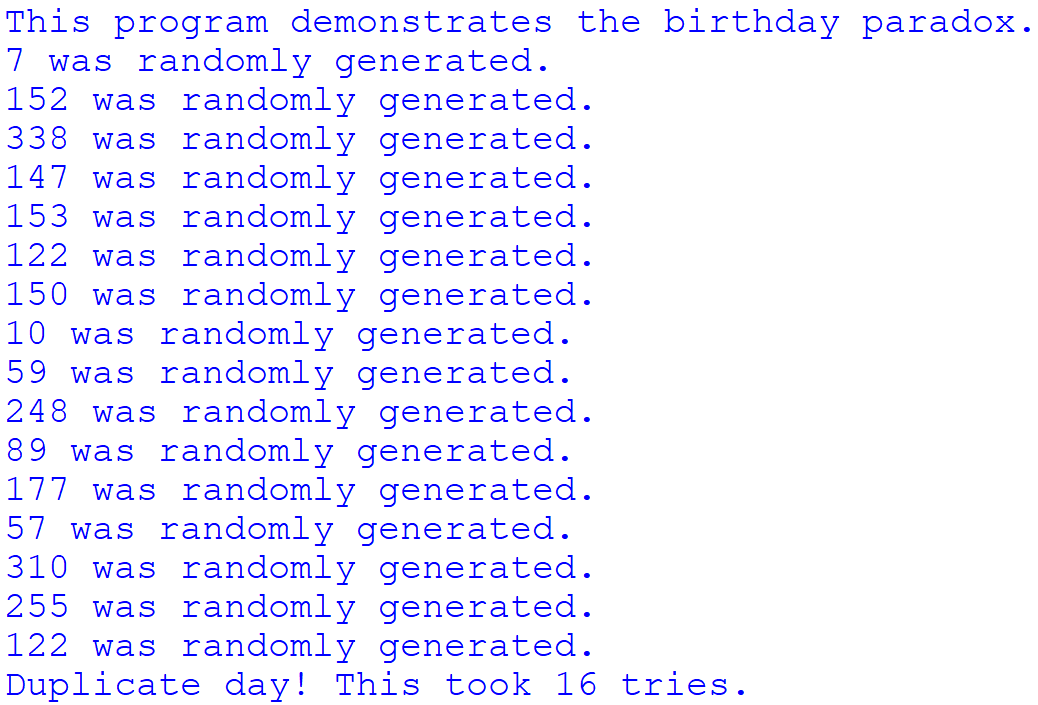
# # # # #

# # # # #

**Activity 9**

Birthday Paradox - ( file name: birthday\_paradox.py )

* The Birthday Paradox asks the question, “If you asked random people their birthday, how many people do you expect to ask before you find two people with the same birthday?”
* Write a program that randomly generates integers between 1 and 365 inclusive, one by one. The program ends when two of the generated values match, and prints out the number of tries it took.
* Sample run:



**Activity 10**

Scope - ( file name: scope.docx )

* Take a look at the following program, and indicate the scope for each of the variables. This includes the global variables x, n and y, and the local variables x, n, p and i.
* For example, the scope for the global variable x is indicated as follows in the diagram. How about the rest?

x

def power(x, n):

p = 1

for i in range(0, n):

p = p \* x

return p

x = 2

n = 10

y = 5

print(power(x, n))

. . .

. . .

**Activity 11**

Printer charges - ( file name: printer\_charges.py )

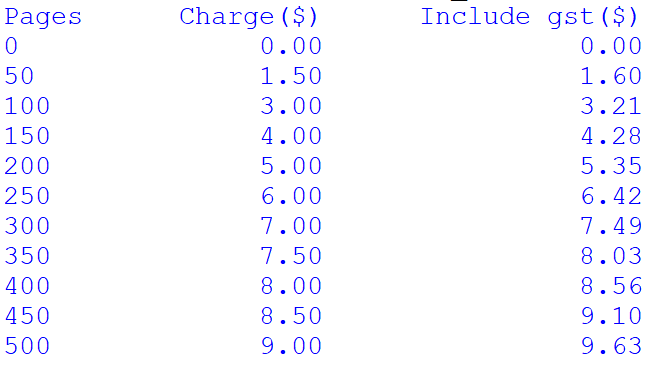
* Consider the following. Computer paper usage from printers is charged at the following rates:

First 100 pages : 3 cents a page

Next 200 pages : 2 cents a page

Over 300 pages : 1 cent a page

* Determine the charge (inclusive of 7% GST) for printing of paper in stacks of 50 pages from 0 to 500 pages.
* Define the following functions in the program:
* calculate\_charge() - takes in the number of pages and return the corresponding charge
* calculate\_gst() - takes in the amount and return the corresponding GST charged
* The expected output is as follows:



**Activity 12**

Obtain Grade – ( file name: obtain\_grade.py )

* Write a function obtain\_grade() that receives a float parameter as a student’s average mark and returns the grade according to the following criteria:

|  |  |
| --- | --- |
| **Average** | **Grade** |
| 84.5 – 100 | A+ |
| 79.5 – 84.5 exclusive | A |
| 74.5 – 79.5 exclusive | B+ |
| 69.5 – 74.5 exclusive | B |
| 64.5 – 69.5 exclusive | C+ |
| 59.5 – 64.5 exclusive | C |
| 54.5 – 59.5 exclusive | D+ |
| 49.5 – 54.5 exclusive | D |
| Below 49.5 | F |

* Incorporate the function in a program that processes a list of students’ averages and display the grade of the averages in a tabular format.
* The list is as follows:

mark\_list = [['Mary', 90.5], ['Charles', 60.4], ['John', 70.5], ['Javier', 32.0], ['Luke', 46.7]]

* Expected output:

Student Name mark Grade

Mary 90.5 A+

Charles 60.4 C

John 70.5 B

Javier 32.0 F

Luke 46.7 F

**Activity 13**

Temperature Conversion – ( file name: convert\_temperature.py )

* Write a function fahr\_to\_cel( ) that returns the Celsius equivalent of a Fahrenheit temperature.

Celsius = 5.0/ 9.0 x (Fahrenheit – 32)

* Write another function cel\_to\_fahr( ) that returns the Fahrenheit equivalent of a Celsius temperature.

Fahrenheit = 9.0/ 5.0 x Celsius + 32

* Using the functions above, write a program that allows the user to choose between two menu options to do either of the two conversions and display the result.
* Sample run:

Temperature Conversion

[1]Fahrenheit to Celsius

[2]Celsius to Fahrenheit

[3]Exit

Please enter your option : 1

Please enter the temperature in Fahrenheit : 100

The temperature in celsius is 37.8 degrees

Temperature Conversion

[1]Fahrenheit to Celsius

[2]Celsius to Fahrenheit

[3]Exit

Please enter your option : 2

Please enter the temperature in Celsius : 36.9

The temperature in fahrenheit is 98.4 degrees

**OPTIONAL**

**Activity 14**

Write an elementary educational program that will allows students to pick the type of arithmetic problem that he/she wishes to practice. An option 1 means addition problems only, option 2 for subtraction problems only, 3 for multiplication problems only, 4 for division problems only, and 5 to randomly intermix problems of these types. The two operands for the arithmetic problems should be randomly generated integer numbers that are less than 10. Example of questions that the program will print is as follows:

How much is 40 \* 30?

How much is 3 – 1?

How much is 1 + 9?

How much is 4 / 2?

The student then types the answer. Your program checks the student’s answers and gives appropriate comment if the answer is correct, or gives other comment to encourage the students to try again if the answer is wrong, and then allows the student to try again until he gets it correct.

To encourage the holding of student’s attention and reducing of student fatigue, the program will randomly generate various comments for each answer and each incorrect answer as follows:

Responses to a correct answer:

Very good!

Excellent!

Nice work!

Keep up the good work!

Responses to an incorrect answer:

No. Please try again.

Wrong. Try once more.

Don’t give up!

No. Keep trying.

In your program, you are to write and call the following functions to enhance the program’s modularity and reusability:

#function menu() prints menu of choices, reads in the user’s choice and returns the #choice chosen by user

int print\_menu()

#function print\_incorrect\_msg() randomly chooses response to incorrect answer.

void print\_incorrect\_msg()

#function print\_correct\_msg() randomly chooses response to correct answer.

void print\_correct\_msg()

You may introduce other functions in your program if needed.