# Grade 11 – Unit 2 – Solutions

*Grade 11 – p 54 – Activity 1 (a)*

Built in functions:

        input()

        float()

        print()

User defined function:

        calcArea()

        rectangleArea()

        triangleArea()

*Grade 11 – p 55 – Activity 1 (b)*

Function definitions:

        Lines 3, 7 & 11

Function calls:

        Lines 18, 19, 21, 22, 24, 25, 26 & 27

*Grade 11 – p 55 – Activity 1 (c)*

data 3 & data 4.

*Grade 11 – p 55 – Activity 1 (d)*

rArea

*Grade 11 – p 55 – Activity 1 (e)*

base, height

| No | Problem description | Function name | Input | Return value or outputs |
|---|---|---|---|---|
| 1 | a function that takes two floating point numbers and adds them | Add() | a(float)<br>b(float) | Sum(float) |
| 2 | a function used to calculate the area<br><br>of circle. | circle_area() | a(float) | area(float) |
| 3 | a function that returns the highest of<br><br>three numbers | highest_numbre | a(float)<br>b(float)<br>c(float) | highest(float) |
| 4 | a function that counts the number of<br><br>words in a file | count_numbers() | words | total_number(int) |
| 5 | a function which checks if the password a user has entered is correct. | check_password() | password(string) | answer(string) |
| 6 | a function to calculate the perimeter<br><br>of a rectangle using the formula p =<br><br>2w+ 2h. | calculate_perimeter() | width(float)<br>height(float) | perimeter(float) |
| 7 | a function to calculate the force<br><br>between two electric charges q1 and<br><br>q2 using Coulomb's Law | calculate_force | q1(float1)<br>q2(float2) | force(float) |
| 8 | a function to check if a number is<br><br>positive, negative or zero | check_number() | num(int) | answer(string) |

| 9 | a function which returns the atmospheric level given the height above sea level | calculate_level() | height(float) | level(float) |
|---|---|---|---|---|

1.

```
1   #Function to print the product of two floats
2   def sum(a, b):
3       product = a + b
4       return product
```

6.

```
1   #Program to print the perimeter of a rectangle
2   def calculatePerimeter (width, height):
3       perimeter = (2 * width) + (2 * height)
4       print("The perimeter of the rectangle is:", perimeter)
```

| No | Problem | Function calling |
|---|---|---|
| 1 | a function circleArea (radius) used to<br><br>calculate the area of a circle from its radius;<br><br>returns the area | Area =<br><br>circleArea(radius) |
| 2 | a function add (x, y) that adds two<br><br>numbers and returns their sum | total = sum(a, b) |
| 3 | a function getHighest (a, b, c) that<br><br>returns the highest of three numbers | highest = getHighest(a, b, c) |
| 4 | a function passCheck (password) to<br><br>check if the password a user has entered is<br><br>correct; returns true if the password is correct<br><br>and false otherwise | ValidPass= passCheck<br><br>(password) |
| 5 | a function graviforce (m1, m2, d)<br><br>that returns the gravitation force between<br><br>two masses m1 and m2 given the distance<br><br>between them | Gforce= graviforce(m1, m2, d) |
| 6 | a function to cuboidVolume (length)<br><br>used to calculate and return the volume of a<br><br>cuboid given its side's length | volume = cuboidVolume(l) |
| 7 | a function getBalance (acoountNumber)<br><br>that returns the balance for a bank account | balance = getBalance(account) |
| 8 | a function getStorage () to check the<br><br>storage space left on your hard drive | storageLeft = getStorage() |

*Grade 11 – p 60 – Activity 4(a)*

We import function from other modules to reuse code that built into Python. Doing this saves time.

*Grade 11 – p 60 – Activity 4(b)*

```
1   #Program to calculate distance between two points
2   from math import sqrt
3
4
5   def distance(x1, y1, x2, y2):
6       dis = sqrt(((x2-x1)**2)+((y2-y1)**2))
7       print(dis)
8
9
10  distance(2, 3, 10, 50)
```

*Grade 11 – p 61 – Activity 4(c)*

```
1   from abc import xyz
```

*Grade 11 – p 61 – Activity 4(d)*

```
1   #Program to print the current date and time
2   import datetime
3   now = datetime.datetime.now()
4   print("Current date and time : ")
5   print(now.strftime("%Y-%m-%d %H:%M:%S"))
```

*Grade 11 – p 61 – Activity 4(e)*

```python
#Program convert angles from degrees to radians and radians
to degrees
from math import *

deg = input("Enter the degree value: ")
rad = input("Enter radian value: ")

deg = float(deg)
deg = radians(deg)
rad = float(rad)
rad = degrees(rad)


print("The degree value converted to radians: ", deg)
print("The radian value converted to degrees: ", rad)
```

*Grade 11 – p 63 –Activity 5 (a)*

```python
#Function that returns volume and surface area of cylinder
def cylinder(r, h):
    vol = pi * r ** 2 * h
    sa = 2 * pi * r * (r + h)
    return vol, sa
```

*Grade 11 – p 63 –Activity 5 (b)*

```python
#Function that sum and difference
def sum_and_difference(x,y):
    product = x + y
    diff = x - y
    return product, diff
```

*Grade 11 – p 65 –Activity 6 (a)*

```python
1  #Function that calculates journey time
2  def calc_journey(d, a=100):
3      time = d/a
4      return time
5
```

*Grade 11 – p 65 –Activity 6 (b)*

```python
1  #Function that calculates total weight
2  def tot_weight(p, w=81):
3      tot = w * p
4      return tot
```

*Grade 11 – p 66 – End of unit Activities (a)*

```python
1   #Function to average calculate average speed
2
3   #Define function with 2 parameters
4   def speed(d, t):
5   #Calculation for speed: speed = distance #divided time
6       s = d / t
7       return s
8
9   #Function to average calculate time
10  def distance(s, t):
11  #Calculation for distance: speed multiplied by time
12      d = s * t
13      return d
14
15  #Function to average calculate time
16  def time(d, s):
17  #Calculation for time: distance divided time
18      t = d / s
19      return t
20
21
22
```

*Grade 11 – p 66 – End of unit Activities (b)*

```
1   #Define function with three parameters
2   def user_info(name, phone, days):
3   #Print the user info in structured way
4       print("Name:", name)
5       print("Phone Number:", phone)
6       print("Number of days:", days)
7
8   #Define function with one default parameter
9   def total_bill(d, r=500):
10  Compute total bill
11      total = r * d
12      return total
13
14  #Call total_bill function and assign the result to tot
15  tot = total_bill(3)
16
17  #Call user_info function with three parameters
18  user_info("Humaid Rashid", 502263489, 5)
19  print("Total bill is:" , tot)
```

*Grade 11 – p 67 – End of unit Activities (c)*

```
1   #Function to convert dirham to USD and Euro
2   def currency_conversion(dirhams):
3   #Doller conversion
4       usd = dirhams / 3.68
5   #Euro conversion
6       euro = dirhams / 4.2
7   #Returning two outputs
8       return usd, euro
9
10  #Dirham to be converted
11  dirham = 100
12
```

```
13    #Call   currency_conversion   function   with   dirham   variable   as
14    #parameter and assigning variables to the 2 ouputs. The order of
      #the variables is important.
15    usd, euro = currency_conversion(dirham)
16

17    print(dirham, "Dirhams converted to Dollars:", usd)
18    print(dirham, "Dirhams converted to Euro:", euro)
19
```

*Grade 11 – p 67 – End of unit Activities (d)*

```
1     #Function to calculate total bill including tax and service charge
2     def total_bill(x):
3         tax = x * 0.05
4         service = x * 0.1
5         tot = x + tax + service
6         return tot
7

8     #Call total_bill function with 244 and assign the result to bill
9     bill = total_bill(244)
10    print("Total bill, including tax and service charge is:", bill)
```

*Grade 11 – p 63 – End of unit Activities (e)*

```
1     #Import the sqrt function from the math module
2     from math import sqrt
3

4     #Define a function that takes four parameters
5     def line_distance(u, d, l, r):
6     #Find y2 but computing the difference between r and l
7         y2 = r - l
8     #Find y2 but computing the difference between u and d
9         x2 = u - d
10    #Calculation for finding line distance
11        dis = sqrt((x2 ** 2) + (y2 ** 2))
12        return dis
```

```
13
14   #Call the line_distance function with 4 parameters and assign
15   #the result to lineDis
16   lineDis = line_distance(3, 1, 2, 5)
17   print("The distance that the robot has moved:", lineDis)
```