

# Activity Task 1: Neural Network, 2021

*Set number: 1*

Instruction: Write solution in Python to given problem. Task 1: 0.3 point, Task 2: 0.3 point, Task 3: 0.4 point,

□

**Task 1** \_\_\_\_\_

Write a function "convert\_temp" that takes one argument "x" representing temperature in Celsius. The function should check if the x is a number (integer or float). If x is the number, the function should convert the temperature to Fahrenheit and print out the converted temperature (Note:  $F = 1.8 * C + 32$ ). In other case, the function should print "Argument x should be float or integer".

**Task 2** \_\_\_\_\_

Write a function "prime\_range" that takes one argument "x". The function should find and print all prime numbers in range from 1 to "x". The function should return the list of all founded prime numbers.

**Task 3** \_\_\_\_\_

Create a class "Calculator" and fallow the steps:

- 1.Its \_\_init\_\_() method should take x and y as arguments. The default values of both x and y are 0. Make sure to set these appropriately in the body of the \_\_init\_\_()method.
- 2.Create a function of "Calculator" class named "set". The function should take x and y as arguments and change values of self.x and self.y.
- 3.Create a function of "Calculator" class named "get". The function have no arguments and should return values of self.x and self.y.
- 4.Create a method of "Calculator" class named "add". The function should return the sum of self.x and self.y.
- 5.Create a method of "Calculator" class named "multiply". The function should return the result of the multiplication of self.x\*self.y.
- 6.Create a method of "Calculator" class named "divide". The function should return the result of the devision of self.x/self.y. If the value of self.y is 0 the function should print "Error: division by zero" and return null.

# Activity Task 1: Neural Network, 2021

*Set number: 2*

Instruction: Write solution in Python to given problem. Task 1: 0.3 point, Task 2: 0.3 point, Task 3: 0.4 point,

□

## Task 1

Define a function "compute\_cost" that takes one argument "food" with default list ["banana", "orange", "apple"]. Define in this function two local dictionaries: stock = "banana": 4, "apple": 0, "orange": 9 and prices = "banana": 3, "apple": 2, "orange": 2.5. The function should loop through each item of "food" and add the price\*stock of present item to calculate total cost. At the end function should print total cost value of products.

## Task 2

Write a function "stats\_list" that takes one argument "x". The x is a list of numbers. The function should calculate and print out: average, maximal and minimal value of list elements. The functions should return list of square roots of "x" elements.

## Task 3

Create a class "File\_operations" and follow the steps:

1. Its \_\_init\_\_() method should takes one argument "fname". The "fname" is a name (or a ptah) of text file with default name "file.txt". Make sure to set these appropriately in the body of the \_\_init\_\_()method.
2. Create a method of "File\_operations" class named "create file" that takes two arguments "fname" and "fcontent". Both arguments of function are strings. The function will update value of self.fname with argument fname, if the fname is not empty. The function will create/write file in path self.fname with contenet described in argument fcontent. If the file exist in this path, the file should be overwritten and print information "File was overwritten".
3. Create a method of "File\_operations" class named "create.info" that takes no arguments. The function should read and display the content of file in path self.fname, and at the end print information about total number of characters in file.



# Activity Task 1: Neural Network, 2021

*Set number: 3*

Instruction: Write solution in Python to given problem. Task 1: 0.3 point, Task 2: 0.3 point, Task 3: 0.4 point,

□

**Task 1** \_\_\_\_\_

Define a function called "remove\_duplicates" that has one argument called "sequence". The "sequence" is the list of integers. The function should print out the number of times each unique number occurs in the list. The function should return "sequence" with only unique numbers (removed duplicates) in ascending order.

**Task 2** \_\_\_\_\_

Define a function called count\_file\_char that has two arguments "fname" and "c". The "fname" is the name of the file to read (create txt file in notepad) and c is the string with only one letter. The function should read a file and count a total number of occurrences of letter c. The function should return string "Total number of letter is " filled with the appropriate values.

**Task 3** \_\_\_\_\_

Create a class "PlotXY" and fallow the steps:

1. Create a method of "PlotXY" class named "generate\_data" with two arguments "xstep" and "xrange". The function should generate data:
  - self.x1 - a sequence of numbers from 0 to xrange with step 0.1\*xstep.
  - self.x2 - a sequence of numbers from 0 to xrange with step xstep.
  - self.y1 - calculate values of sin in range of self.x1
  - self.y2 - calculate values of cosine in range of self.x2
2. Create a method of "PlotXY" class named "plot1" with no arguments. The function should plot both sin (defined by self.x1 and self.y1) and cosine (defined by self.x2 and self.y2) in the same figure. The sinues should draw with red solid line and cosine with blue solid line. The plot should have:
  - title - defined by argument "title".
  - x axis label - should be "x".
  - y axis label - should be "y".
  - legend = should be 'sin' and 'cosine'.
3. Create a method of "PlotXY" class named "plot2" with no arguments. The function should create 2 subplots of sin (defined by self.x1 and self.y1) and cosine (defined by self.x2 and self.y2). The sinues should draw with red dashed line and cosine with blue dashed line. The each subplot should have:
  - x axis label - should be "x".
  - y axis label - should be "y".
  - grid on

# Activity Task 1: Neural Network, 2021

*Set number: 4*

Instruction: Write solution in Python to given problem. Task 1: 0.3 point, Task 2: 0.3 point, Task 3: 0.4 point,

□

**Task 1** \_\_\_\_\_

Write a function "convert\_temp" that takes one argument "x" representing temperature in Celsius. The function should check if the x is a number (integer or float). If x is the number, the function should convert the temperature to Fahrenheit and print out the converted temperature (Note:  $F = 1.8 * C + 32$ ). In other case, the function should print "Argument x should be float or integer".

**Task 2** \_\_\_\_\_

Define a function called crop\_image that has five arguments called "fname", "x", "y", "w", "h". The "fname" is the path and name of the image file, (x, y, w, h) are parameters of rectangle, where (x, y) are coordinates of top left corner of rectangle, and (w, h) are width and height of rectangle. The (x, y, w, h) describe the area of image "fname" to be cropped. If the parameters of area to be cropped are outside the range of image, they should be limited to the image dimensions (e.g. if x = -10 it should be corrected to 0). The function should return cropped image.

**Task 3** \_\_\_\_\_

Create a class "Calculator" and fallow the steps:

- 1.Its \_\_init\_\_() method should take x and y as arguments. The default values of both x and y are 0. Make sure to set these appropriately in the body of the \_\_init\_\_()method.
- 2.Create a function of "Calculator" class named "set". The function should take x and y as arguments and change values of self.x and self.y.
- 3.Create a function of "Calculator" class named "get". The function have no arguments and should return values of self.x and self.y.
- 4.Create a method of "Calculator" class named "add". The function should return the sum of self.x and self.y.
- 5.Create a method of "Calculator" class named "multiply". The function should return the result of the multiplication of self.x\*self.y.
- 6.Create a method of "Calculator" class named "divide". The function should return the result of the devision of self.x/self.y. If the value of self.y is 0 the function should print "Error: division by zero" and return null.



# Activity Task 1: Neural Network, 2021

*Set number: 5*

Instruction: Write solution in Python to given problem. Task 1: 0.3 point, Task 2: 0.3 point, Task 3: 0.4 point,

□

**Task 1** \_\_\_\_\_

Write a function "prime\_range" that takes one argument "x". The function should find and print all prime numbers in range from 1 to "x". The function should return the list of all founded prime numbers.

**Task 2** \_\_\_\_\_

Define a function "compute\_cost" that takes one argument "food" with default list ["banana", "orange", "apple"]. Define in this function two local dictionaries: stock = "banana": 4, "apple": 0, "orange": 9 and prices = "banana": 3, "apple": 2, "orange": 2.5. The function should loop through each item of "food" and add the price\*stock of present item to calculate total cost. At the end function should print total cost value of products.

**Task 3** \_\_\_\_\_

Create a class "File\_operations" and fallow the steps:

- 1.Its \_\_init\_\_() method should takes one argument "fname". The "fname" is a name (or a ptah) of text file with default name "file.txt". Make sure to set these appropriately in the body of the \_\_init\_\_()method.
- 2.Create a method of "File\_operations" class named "create file" that takes two arguments "fname" and "fcontent". Both arguments of function are strings. The function will update value of self.fname with argument fname, if the fname is not empty. The function will create/write file in path self.fname with contenet described in argument fcontent. If the file exist in this path, the file should be overwritten and print information "File was overwritten".
- 3.Create a method of "File\_operations" class named "create.info" that takes no arguments. The function should read and display the content of file in path self.fname, and at the end print information about total number of characters in file.

# Activity Task 1: Neural Network, 2021

*Set number: 6*

Instruction: Write solution in Python to given problem. Task 1: 0.3 point, Task 2: 0.3 point, Task 3: 0.4 point,

□

**Task 1** \_\_\_\_\_

Write a function "stats\_list" that takes one argument "x". The x is a list of numbers. The function should calculate and print out: average, maximal and minimal value of list elements. The functions should return list of square roots of "x" elements.

**Task 2** \_\_\_\_\_

Define a function called "remove\_duplicates" that has one argument called "sequence". The "sequence" is the list of integers. The function should print out the number of times each unique number occurs in the list. The function should return "sequence" with only unique numbers (removed duplicates) in ascending order.

**Task 3** \_\_\_\_\_

Create a class "PlotXY" and fallow the steps:

1. Create a method of "PlotXY" class named "generate\_data" with two arguments "xstep" and "xrange". The function should generate data:
  - self.x1 - a sequence of numbers from 0 to xrange with step 0.1\*xstep.
  - self.x2 - a sequence of numbers from 0 to xrange with step xstep.
  - self.y1 - calculate values of sin in range of self.x1
  - self.y2 - calculate values of cosine in range of self.x2
2. Create a method of "PlotXY" class named "plot1" with no arguments. The function should plot both sin (defined by self.x1 and self.y1) and cosine (defined by self.x2 and self.y2) in the same figure. The sinues should draw with red solid line and cosine with blue solid line. The plot should have:
  - title - defined by argument "title".
  - x axis label - should be "x".
  - y axis label - should be "y".
  - legend = should be 'sin' and 'cosine'.
3. Create a method of "PlotXY" class named "plot2" with no arguments. The function should create 2 subplots of sin (defined by self.x1 and self.y1) and cosine (defined by self.x2 and self.y2). The sinues should draw with red dashed line and cosine with blue dashed line. The each subplot should have:
  - x axis label - should be "x".
  - y axis label - should be "y".
  - grid on



# Activity Task 1: Neural Network, 2021

*Set number: 7*

Instruction: Write solution in Python to given problem. Task 1: 0.3 point, Task 2: 0.3 point, Task 3: 0.4 point,

□

**Task 1** \_\_\_\_\_

Define a function called count\_file\_char that has two arguments "fname" and "c". The "fname" is the name of the file to read (create txt file in notepad) and c is the string with only one letter. The function should read a file and count a total number of occurrences of letter c. The function should return string "Total number of letter is " filled with the appropriate values.

**Task 2** \_\_\_\_\_

Define a function called crop\_image that has five arguments called "fname", "x", "y", "w", "h". The "fname" is the path and name of the image file, (x, y, w, h) are parameters of rectangle, where (x, y) are coordinates of top left corner of rectangle, and (w, h) are width and height of rectangle. The (x, y, w, h) describe the area of image "fname" to be cropped. If the parameters of area to be cropped are outside the range of image, they should be limited to the image dimensions (e.g. if x = -10 it should be corrected to 0). The function should return cropped image.

**Task 3** \_\_\_\_\_

Create a class "Calculator" and fallow the steps:

- 1.Its \_\_init\_\_() method should take x and y as arguments. The default values of both x and y are 0. Make sure to set these appropriately in the body of the \_\_init\_\_()method.
- 2.Create a function of "Calculator" class named "set". The function should take x and y as arguments and change values of self.x and self.y.
- 3.Create a function of "Calculator" class named "get". The function have no arguments and should return values of self.x and self.y.
- 4.Create a method of "Calculator" class named "add". The function should return the sum of self.x and self.y.
- 5.Create a method of "Calculator" class named "multiply". The function should return the result of the multiplication of self.x\*self.y.
- 6.Create a method of "Calculator" class named "divide". The function should return the result of the devision of self.x/self.y. If the value of self.y is 0 the function should print "Error: division by zero" and return null.

# Activity Task 1: Neural Network, 2021

*Set number: 8*

Instruction: Write solution in Python to given problem. Task 1: 0.3 point, Task 2: 0.3 point, Task 3: 0.4 point,

□

## Task 1

Write a function "convert\_temp" that takes one argument "x" representing temperature in Celsius. The function should check if the x is a number (integer or float). If x is the number, the function should convert the temperature to Fahrenheit and print out the converted temperature (Note:  $F = 1.8 * C + 32$ ). In other case, the function should print "Argument x should be float or integer".

## Task 2

Write a function "prime\_range" that takes one argument "x". The function should find and print all prime numbers in range from 1 to "x". The function should return the list of all founded prime numbers.

## Task 3

Create a class "File\_operations" and fallow the steps:

1. Its \_\_init\_\_() method should takes one argument "fname". The "fname" is a name (or a ptah) of text file with default name "file.txt". Make sure to set these appropriately in the body of the \_\_init\_\_()method.
2. Create a method of "File\_operations" class named "create file" that takes two arguments "fname" and "fcontent". Both arguments of function are strings. The function will update value of self.fname with argument fname, if the fname is not empty. The function will create/write file in path self.fname with contenet described in argument fcontent. If the file exist in this path, the file should be overwritten and print information "File was overwritten".
3. Create a method of "File\_operations" class named "create.info" that takes no arguments. The function should read and display the content of file in path self.fname, and at the end print information about total number of characters in file.



# Activity Task 1: Neural Network, 2021

*Set number: 9*

Instruction: Write solution in Python to given problem. Task 1: 0.3 point, Task 2: 0.3 point, Task 3: 0.4 point,

□

**Task 1** \_\_\_\_\_

Define a function "compute\_cost" that takes one argument "food" with default list ["banana", "orange", "apple"]. Define in this function two local dictionaries: stock = "banana": 4, "apple": 0, "orange": 9 and prices = "banana": 3, "apple": 2, "orange": 2.5. The function should loop through each item of "food" and add the price\*stock of present item to calculate total cost. At the end function should print total cost value of products.

**Task 2** \_\_\_\_\_

Write a function "stats\_list" that takes one argument "x". The x is a list of numbers. The function should calculate and print out: average, maximal and minimal value of list elements. The functions should return list of square roots of "x" elements.

**Task 3** \_\_\_\_\_

Create a class "PlotXY" and fallow the steps:

1. Create a method of "PlotXY" class named "generate\_data" with two arguments "xstep" and "xrange". The function should generate data:
  - self.x1 - a sequence of numbers from 0 to xrange with step 0.1\*xstep.
  - self.x2 - a sequence of numbers from 0 to xrange with step xstep.
  - self.y1 - calculate values of sin in range of self.x1
  - self.y2 - calculate values of cosine in range of self.x2
2. Create a method of "PlotXY" class named "plot1" with no arguments. The function should plot both sin (defined by self.x1 and self.y1) and cosine (defined by self.x2 and self.y2) in the same figure. The sinues should draw with red solid line and cosine with blue solid line. The plot should have:
  - title - defined by argument "title".
  - x axis label - should be "x".
  - y axis label - should be "y".
  - legend = should be 'sin' and 'cosine'.
3. Create a method of "PlotXY" class named "plot2" with no arguments. The function should create 2 subplots of sin (defined by self.x1 and self.y1) and cosine (defined by self.x2 and self.y2). The sinues should draw with red dashed line and cosine with blue dashed line. The each subplot should have:
  - x axis label - should be "x".
  - y axis label - should be "y".
  - grid on

# Activity Task 1: Neural Network, 2021

*Set number: 10*

Instruction: Write solution in Python to given problem. Task 1: 0.3 point, Task 2: 0.3 point, Task 3: 0.4 point,

□

**Task 1** \_\_\_\_\_

Define a function called "remove\_duplicates" that has one argument called "sequence". The "sequence" is the list of integers. The function should print out the number of times each unique number occurs in the list. The function should return "sequence" with only unique numbers (removed duplicates) in ascending order.

**Task 2** \_\_\_\_\_

Define a function called count\_file\_char that has two arguments "fname" and "c". The "fname" is the name of the file to read (create txt file in notepad) and c is the string with only one letter. The function should read a file and count a total number of occurrences of letter c. The function should return string "Total number of letter is " filled with the appropriate values.

**Task 3** \_\_\_\_\_

Create a class "Calculator" and fallow the steps:

- 1.Its \_\_init\_\_() method should take x and y as arguments. The default values of both x and y are 0. Make sure to set these appropriately in the body of the \_\_init\_\_()method.
- 2.Create a function of "Calculator" class named "set". The function should take x and y as arguments and change values of self.x and self.y.
- 3.Create a function of "Calculator" class named "get". The function have no arguments and should return values of self.x and self.y.
- 4.Create a method of "Calculator" class named "add". The function should return the sum of self.x and self.y.
- 5.Create a method of "Calculator" class named "multiply". The function should return the result of the multiplication of self.x\*self.y.
- 6.Create a method of "Calculator" class named "divide". The function should return the result of the devision of self.x/self.y. If the value of self.y is 0 the function should print "Error: division by zero" and return null.



# Activity Task 1: Neural Network, 2021

*Set number: 11*

Instruction: Write solution in Python to given problem. Task 1: 0.3 point, Task 2: 0.3 point, Task 3: 0.4 point,

□

**Task 1** \_\_\_\_\_

Write a function "convert\_temp" that takes one argument "x" representing temperature in Celsius. The function should check if the x is a number (integer or float). If x is the number, the function should convert the temperature to Fahrenheit and print out the converted temperature (Note:  $F = 1.8 * C + 32$ ). In other case, the function should print "Argument x should be float or integer".

**Task 2** \_\_\_\_\_

Define a function called crop\_image that has five arguments called "fname", "x", "y", "w", "h". The "fname" is the path and name of the image file, (x, y, w, h) are parameters of rectangle, where (x, y) are coordinates of top left corner of rectangle, and (w, h) are width and height of rectangle. The (x, y, w, h) describe the area of image "fname" to be cropped. If the parameters of area to be cropped are outside the range of image, they should be limited to the image dimensions (e.g. if x = -10 it should be corrected to 0). The function should return cropped image.

**Task 3** \_\_\_\_\_

Create a class "File\_operations" and fallow the steps:

- 1.Its \_\_init\_\_() method should takes one argument "fname". The "fname" is a name (or a ptah) of text file with default name "file.txt". Make sure to set these appropriately in the body of the \_\_init\_\_()method.
- 2.Create a method of "File\_operations" class named "create file" that takes two arguments "fname" and "fcontent". Both arguments of function are strings. The function will update value of self.fname with argument fname, if the fname is not empty. The function will create/write file in path self.fname with contenet described in argument fcontent. If the file exist in this path, the file should be overwritten and print information "File was overwritten".
- 3.Create a method of "File\_operations" class named "create\_info" that takes no arguments. The function should read and display the content of file in path self.fname, and at the end print information about total number of characters in file.

# Activity Task 1: Neural Network, 2021

*Set number: 12*

Instruction: Write solution in Python to given problem. Task 1: 0.3 point, Task 2: 0.3 point, Task 3: 0.4 point,

□

## Task 1

Write a function "prime\_range" that takes one argument "x". The function should find and print all prime numbers in range from 1 to "x". The function should return the list of all founded prime numbers.

## Task 2

Define a function "compute\_cost" that takes one argument "food" with default list ["banana", "orange", "apple"]. Define in this function two local dictionaries: stock = "banana": 4, "apple": 0, "orange": 9 and prices = "banana": 3, "apple": 2, "orange": 2.5. The function should loop through each item of "food" and add the price\*stock of present item to calculate total cost. At the end function should print total cost value of products.

## Task 3

Create a class "PlotXY" and fallow the steps:

- Create a method of "PlotXY" class named "generate\_data" with two arguments "xstep" and "xrange". The function should generate data:
  - self.x1 - a sequence of numbers from 0 to xrange with step 0.1\*xstep.
  - self.x2 - a sequence of numbers from 0 to xrange with step xstep.
  - self.y1 - calculate values of sin in range of self.x1
  - self.y2 - calculate values of cosine in range of self.x2
- Create a method of "PlotXY" class named "plot1" with no arguments. The function should plot both sin (defined by self.x1 and self.y1) and cosine (defined by self.x2 and self.y2) in the same figure. The sinues should draw with red solid line and cosine with blue solid line. The plot should have:
  - title - defined by argument "title".
  - x axis label - should be "x".
  - y axis label - should be "y".
  - legend = should be 'sin' and 'cosine'.
- Create a method of "PlotXY" class named "plot2" with no arguments. The function should create 2 subplots of sin (defined by self.x1 and self.y1) and cosine (defined by self.x2 and self.y2). The sinues should draw with red dashed line and cosine with blue dashed line. The each subplot should have:
  - x axis label - should be "x".
  - y axis label - should be "y".
  - grid on



# Activity Task 1: Neural Network, 2021

*Set number: 13*

Instruction: Write solution in Python to given problem. Task 1: 0.3 point, Task 2: 0.3 point, Task 3: 0.4 point,

□

**Task 1** \_\_\_\_\_

Write a function "stats\_list" that takes one argument "x". The x is a list of numbers. The function should calculate and print out: average, maximal and minimal value of list elements. The functions should return list of square roots of "x" elements.

**Task 2** \_\_\_\_\_

Define a function called "remove\_duplicates" that has one argument called "sequence". The "sequence" is the list of integers. The function should print out the number of times each unique number occurs in the list. The function should return "sequence" with only unique numbers (removed duplicates) in ascending order.

**Task 3** \_\_\_\_\_

Create a class "Calculator" and fallow the steps:

- 1.Its \_\_init\_\_() method should take x and y as arguments. The default values of both x and y are 0. Make sure to set these appropriately in the body of the \_\_init\_\_()method.
- 2.Create a function of "Calculator" class named "set". The function should take x and y as arguments and change values of self.x and self.y.
- 3.Create a function of "Calculator" class named "get". The function have no arguments and should return values of self.x and self.y.
- 4.Create a method of "Calculator" class named "add". The function should return the sum of self.x and self.y.
- 5.Create a method of "Calculator" class named "multiply". The function should return the result of the multiplication of self.x\*self.y.
- 6.Create a method of "Calculator" class named "divide". The function should return the result of the devision of self.x/self.y. If the value of self.y is 0 the function should print "Error: division by zero" and return null.

# Activity Task 1: Neural Network, 2021

*Set number: 14*

Instruction: Write solution in Python to given problem. Task 1: 0.3 point, Task 2: 0.3 point, Task 3: 0.4 point,

□

## Task 1

Define a function called `count_file_char` that has two arguments `"fname"` and `"c"`. The `"fname"` is the name of the file to read (create txt file in notepad) and `c` is the string with only one letter. The function should read a file and count a total number of occurrences of letter `c`. The function should return string `"Total number of letter is "` filled with the appropriate values.

## Task 2

Define a function called `crop_image` that has five arguments called `"fname"`, `"x"`, `"y"`, `"w"`, `"h"`. The `"fname"` is the path and name of the image file, `(x, y, w, h)` are parameters of rectangle, where `(x, y)` are coordinates of top left corner of rectangle, and `(w, h)` are width and height of rectangle. The `(x, y, w, h)` describe the area of image `"fname"` to be cropped. If the parameters of area to be cropped are outside the range of image, they should be limited to the image dimensions (e.g. if `x = -10` it should be corrected to 0). The function should return cropped image.

## Task 3

Create a class `"PlotXY"` and follow the steps:

- Create a method of `"PlotXY"` class named `"generate_data"` with two arguments `"xstep"` and `"xrange"`. The function should generate data:
  - `self.x1` - a sequence of numbers from 0 to `xrange` with step `0.1*xstep`.
  - `self.x2` - a sequence of numbers from 0 to `xrange` with step `xstep`.
  - `self.y1` - calculate values of sin in range of `self.x1`
  - `self.y2` - calculate values of cosine in range of `self.x2`
- Create a method of `"PlotXY"` class named `"plot1"` with no arguments. The function should plot both sin (defined by `self.x1` and `self.y1`) and cosine (defined by `self.x2` and `self.y2`) in the same figure. The sinues should draw with red solid line and cosine with blue solid line. The plot should have:
  - `title` - defined by argument `"title"`.
  - `x` axis label - should be `"x"`.
  - `y` axis label - should be `"y"`.
  - `legend` = should be `'sin'` and `'cosine'`.
- Create a method of `"PlotXY"` class named `"plot2"` with no arguments. The function should create 2 subplots of sin (defined by `self.x1` and `self.y1`) and cosine (defined by `self.x2` and `self.y2`). The sinues should draw with red dashed line and cosine with blue dashed line. The each subplot should have:
  - `x` axis label - should be `"x"`.
  - `y` axis label - should be `"y"`.
  - `grid` on



# Activity Task 1: Neural Network, 2021

*Set number: 15*

Instruction: Write solution in Python to given problem. Task 1: 0.3 point, Task 2: 0.3 point, Task 3: 0.4 point,

□

## Task 1

Write a function "convert\_temp" that takes one argument "x" representing temperature in Celsius. The function should check if the x is a number (integer or float). If x is the number, the function should convert the temperature to Fahrenheit and print out the converted temperature (Note:  $F = 1.8 * C + 32$ ). In other case, the function should print "Argument x should be float or integer".

## Task 2

Write a function "prime\_range" that takes one argument "x". The function should find and print all prime numbers in range from 1 to "x". The function should return the list of all founded prime numbers.

## Task 3

Create a class "File\_operations" and fallow the steps:

1. Its \_\_init\_\_() method should takes one argument "fname". The "fname" is a name (or a ptah) of text file with default name "file.txt". Make sure to set these appropriately in the body of the \_\_init\_\_()method.
2. Create a method of "File\_operations" class named "create file" that takes two arguments "fname" and "fcontent". Both arguments of function are strings. The function will update value of self.fname with argument fname, if the fname is not empty. The function will create/write file in path self.fname with contenet described in argument fcontent. If the file exist in this path, the file should be overwritten and print information "File was overwritten".
3. Create a method of "File\_operations" class named "create.info" that takes no arguments. The function should read and display the content of file in path self.fname, and at the end print information about total number of characters in file.

# Activity Task 1: Neural Network, 2021

*Set number: 16*

Instruction: Write solution in Python to given problem. Task 1: 0.3 point, Task 2: 0.3 point, Task 3: 0.4 point,

□

## Task 1

Define a function "compute\_cost" that takes one argument "food" with default list ["banana", "orange", "apple"]. Define in this function two local dictionaries: stock = "banana": 4, "apple": 0, "orange": 9 and prices = "banana": 3, "apple": 2, "orange": 2.5. The function should loop through each item of "food" and add the price\*stock of present item to calculate total cost. At the end function should print total cost value of products.

## Task 2

Write a function "stats\_list" that takes one argument "x". The x is a list of numbers. The function should calculate and print out: average, maximal and minimal value of list elements. The functions should return list of square roots of "x" elements.

## Task 3

Create a class "Calculator" and follow the steps:

1. Its `__init__()` method should take x and y as arguments. The default values of both x and y are 0. Make sure to set these appropriately in the body of the `__init__()` method.
2. Create a function of "Calculator" class named "set". The function should take x and y as arguments and change values of self.x and self.y.
3. Create a function of "Calculator" class named "get". The function have no arguments and should return values of self.x and self.y.
4. Create a method of "Calculator" class named "add". The function should return the sum of self.x and self.y.
5. Create a method of "Calculator" class named "multiply". The function should return the result of the multiplication of self.x\*self.y.
6. Create a method of "Calculator" class named "divide". The function should return the result of the devision of self.x/self.y. If the value of self.y is 0 the function should print "Error: division by zero" and return null.



# Activity Task 1: Neural Network, 2021

*Set number: 17*

Instruction: Write solution in Python to given problem. Task 1: 0.3 point, Task 2: 0.3 point, Task 3: 0.4 point,

□

**Task 1** \_\_\_\_\_

Define a function called "remove\_duplicates" that has one argument called "sequence". The "sequence" is the list of integers. The function should print out the number of times each unique number occurs in the list. The function should return "sequence" with only unique numbers (removed duplicates) in ascending order.

**Task 2** \_\_\_\_\_

Define a function called count\_file\_char that has two arguments "fname" and "c". The "fname" is the name of the file to read (create txt file in notepad) and c is the string with only one letter. The function should read a file and count a total number of occurrences of letter c. The function should return string "Total number of letter is " filled with the appropriate values.

**Task 3** \_\_\_\_\_

Create a class "PlotXY" and fallow the steps:

- Create a method of "PlotXY" class named "generate\_data" with two arguments "xstep" and "xrange". The function should generate data:
  - self.x1 - a sequence of numbers from 0 to xrange with step 0.1\*xstep.
  - self.x2 - a sequence of numbers from 0 to xrange with step xstep.
  - self.y1 - calculate values of sin in range of self.x1
  - self.y2 - calculate values of cosine in range of self.x2
- Create a method of "PlotXY" class named "plot1" with no arguments. The function should plot both sin (defined by self.x1 and self.y1) and cosine (defined by self.x2 and self.y2) in the same figure. The sinues should draw with red solid line and cosine with blue solid line. The plot should have:
  - title - defined by argument "title".
  - x axis label - should be "x".
  - y axis label - should be "y".
  - legend = should be 'sin' and 'cosine'.
- Create a method of "PlotXY" class named "plot2" with no arguments. The function should create 2 subplots of sin (defined by self.x1 and self.y1) and cosine (defined by self.x2 and self.y2). The sinues should draw with red dashed line and cosine with blue dashed line. The each subplot should have:
  - x axis label - should be "x".
  - y axis label - should be "y".
  - grid on

# Activity Task 1: Neural Network, 2021

*Set number: 18*

Instruction: Write solution in Python to given problem. Task 1: 0.3 point, Task 2: 0.3 point, Task 3: 0.4 point,

□

## Task 1

Write a function "convert\_temp" that takes one argument "x" representing temperature in Celsius. The function should check if the x is a number (integer or float). If x is the number, the function should convert the temperature to Fahrenheit and print out the converted temperature (Note:  $F = 1.8 * C + 32$ ). In other case, the function should print "Argument x should be float or integer".

## Task 2

Define a function called crop\_image that has five arguments called "fname", "x", "y", "w", "h". The "fname" is the path and name of the image file, (x, y, w, h) are parameters of rectangle, where (x, y) are coordinates of top left corner of rectangle, and (w, h) are width and height of rectangle. The (x, y, w, h) describe the area of image "fname" to be cropped. If the parameters of area to be cropped are outside the range of image, they should be limited to the image dimensions (e.g. if x = -10 it should be corrected to 0). The function should return cropped image.

## Task 3

Create a class "File\_operations" and fallow the steps:

- 1.Its \_\_init\_\_() method should takes one argument "fname". The "fname" is a name (or a ptah) of text file with default name "file.txt". Make sure to set these appropriately in the body of the \_\_init\_\_()method.
- 2.Create a method of "File\_operations" class named "create file" that takes two arguments "fname" and "fcontent". Both arguments of function are strings. The function will update value of self.fname with argument fname, if the fname is not empty. The function will create/write file in path self.fname with contenet described in argument fcontent. If the file exist in this path, the file should be overwritten and print information "File was overwritten".
- 3.Create a method of "File\_operations" class named "create\_info" that takes no arguments. The function should read and display the content of file in path self.fname, and at the end print information about total number of characters in file.



# Activity Task 1: Neural Network, 2021

*Set number: 19*

Instruction: Write solution in Python to given problem. Task 1: 0.3 point, Task 2: 0.3 point, Task 3: 0.4 point,

□

## Task 1

Write a function "prime\_range" that takes one argument "x". The function should find and print all prime numbers in range from 1 to "x". The function should return the list of all founded prime numbers.

## Task 2

Define a function called "remove\_duplicates" that has one argument called "sequence". The "sequence" is the list of integers. The function should print out the number of times each unique number occurs in the list. The function should return "sequence" with only unique numbers (removed duplicates) in ascending order.

## Task 3

Create a class "Calculator" and fallow the steps:

- 1.Its \_\_init\_\_() method should take x and y as arguments. The default values of both x and y are 0. Make sure to set these appropriately in the body of the \_\_init\_\_()method.
- 2.Create a function of "Calculator" class named "set". The function should take x and y as arguments and change values of self.x and self.y.
- 3.Create a function of "Calculator" class named "get". The function have no arguments and should return values of self.x and self.y.
- 4.Create a method of "Calculator" class named "add". The function should return the sum of self.x and self.y.
- 5.Create a method of "Calculator" class named "multiply". The function should return the result of the multiplication of self.x\*self.y.
- 6.Create a method of "Calculator" class named "divide". The function should return the result of the devision of self.x/self.y. If the value of self.y is 0 the function should print "Error: division by zero" and return null.

# Activity Task 1: Neural Network, 2021

*Set number: 20*

Instruction: Write solution in Python to given problem. Task 1: 0.3 point, Task 2: 0.3 point, Task 3: 0.4 point,

□

## Task 1

Define a function called "remove\_duplicates" that has one argument called "sequence". The "sequence" is the list of integers. The function should print out the number of times each unique number occurs in the list. The function should return "sequence" with only unique numbers (removed duplicates) in ascending order.

## Task 2

Define a function called count\_file\_char that has two arguments "fname" and "c". The "fname" is the name of the file to read (create txt file in notepad) and c is the string with only one letter. The function should read a file and count a total number of occurrences of letter c. The function should return string "Total number of letter is " filled with the appropriate values.

## Task 3

Create a class "PlotXY" and fallow the steps:

- Create a method of "PlotXY" class named "generate\_data" with two arguments "xstep" and "xrange". The function should generate data:
  - self.x1 - a sequence of numbers from 0 to xrange with step 0.1\*xstep.
  - self.x2 - a sequence of numbers from 0 to xrange with step xstep.
  - self.y1 - calculate values of sin in range of self.x1
  - self.y2 - calculate values of cosine in range of self.x2
- Create a method of "PlotXY" class named "plot1" with no arguments. The function should plot both sin (defined by self.x1 and self.y1) and cosine (defined by self.x2 and self.y2) in the same figure. The sinues should draw with red solid line and cosine with blue solid line. The plot should have:
  - title - defined by argument "title".
  - x axis label - should be "x".
  - y axis label - should be "y".
  - legend = should be 'sin' and 'cosine'.
- Create a method of "PlotXY" class named "plot2" with no arguments. The function should create 2 subplots of sin (defined by self.x1 and self.y1) and cosine (defined by self.x2 and self.y2). The sinues should draw with red dashed line and cosine with blue dashed line. The each subplot should have:
  - x axis label - should be "x".
  - y axis label - should be "y".
  - grid on



# Activity Task 1: Neural Network, 2021

*Set number: 21*

Instruction: Write solution in Python to given problem. Task 1: 0.3 point, Task 2: 0.3 point, Task 3: 0.4 point,

□

**Task 1** \_\_\_\_\_

Write a function "convert\_temp" that takes one argument "x" representing temperature in Celsius. The function should check if the x is a number (integer or float). If x is the number, the function should convert the temperature to Fahrenheit and print out the converted temperature (Note:  $F = 1.8 * C + 32$ ). In other case, the function should print "Argument x should be float or integer".

**Task 2** \_\_\_\_\_

Define a function called "remove\_duplicates" that has one argument called "sequence". The "sequence" is the list of integers. The function should print out the number of times each unique number occurs in the list. The function should return "sequence" with only unique numbers (removed duplicates) in ascending order.

**Task 3** \_\_\_\_\_

Create a class "Calculator" and fallow the steps:

1. Its \_\_init\_\_() method should take x and y as arguments. The default values of both x and y are 0. Make sure to set these appropriately in the body of the \_\_init\_\_()method.
2. Create a function of "Calculator" class named "set". The function should take x and y as arguments and change values of self.x and self.y.
3. Create a function of "Calculator" class named "get". The function have no arguments and should return values of self.x and self.y.
4. Create a method of "Calculator" class named "add". The function should return the sum of self.x and self.y.
5. Create a method of "Calculator" class named "multiply". The function should return the result of the multiplication of self.x\*self.y.
6. Create a method of "Calculator" class named "divide". The function should return the result of the devision of self.x/self.y. If the value of self.y is 0 the function should print "Error: division by zero" and return null.