1 Python language exercises

1. (5 points) Consider the following expression, intended to print the square root of 16:

   ```
   pow(16,(1/2))
   ```

   What is the result of this expression? How should it be changed, still using `pow`, to yield the correct answer?

2. (5 points) Define the variables `x` and `y` as lists of numbers, and `z` as a tuple.

   ```
   x=[1, 2, 3, 4, 5]
   y=[11, 12, 13, 14, 15]
   z=(21, 22, 23, 24, 25)
   ```

   (a) What is the value of `3*x`?

   (b) What is the value of `x+y`?

   (c) What is the value of `x-y`?

   (d) What is the value of `x[1]`?

   (e) What is the value of `x[0]`?

   (f) What is the value of `x[-1]`?

   (g) What is the value of `x[:]`?

   (h) What is the value of `x[2:4]`?

   (i) What is the value of `x[1:4:2]`?

   (j) What is the value of `x[:2]`?

   (k) What is the value of `x[::2]`?

   (l) What is the result of the following two expressions?

   ```
   x[3]=8
   print x
   ```

   (m) What is the result of the above pair of expressions if the list `x` were replaced with the tuple `z`?

3. (5 points) Define the variable `s` as the string `s="abcde"`.

   (a) What is the value of `3*x`?

   (b) What is the value of `x[1]`?

   (c) What is the value of `x[-1]`?

   (d) What is the value of `x[::2]`?

4. (5 points) Write a program to find those numbers ¡=100 that are equal to the sum of their factors.

5. (5 points) Define a "Big number" as a list of digits. Write a program whose first non-comment lines are two 20-digit numbers
`x=[3,1,4,1,5,9,2,6,5,3,5,8,9,7,9,3,2,3,8,5]`
`y=[2,7,1,8,2,8,1,8,2,8,4,5,9,0,4,5,5,3,4,9]`
Write a program that:

   (a) Finds the sum (x+y) *considered as 20-digit "big numbers"* and prints it as a list of digits.

   (b) Finds the product 15*x *considered as a product of "big numbers"* and prints it as a list of digits.

   Your program should be general enough that if x and y are changed to be 30-digit numbers, your program would still work correctly.

6. (8 points) Write a program to do the following tasks:

   (a) Define a function named `dif2` that accepts an integer N as input parameter and constructs and returns an $N \times N$ two-dimensional `numpy` array `A`, with the value -2.0 on the main diagonal and the value +1.0 on the super-diagonal and the sub-diagonal.

   (b) For N=10, construct a one-dimensional array `b` of length N filled with zeros except that the first element is 1.0 and the last element is -N. For N=10, solve the system $Ax = b$ for x.

   (c) For N=20, construct a one-dimensional array `c` of length N, filled with random numbers. For A from the `dif2` function, Solve the system $Ay = c$ for y and then confirm that the solution you found is approximately correct by computing the relative norm of the residual error, $\|Ay - c\|/\|c\|$. This value should be no larger than 10^{-12}.