7. Provide the expected value for x in the following Python 3 expressions

```
x=[0]*3
```

[0] is a list of containing the element 0. We multiply this element per 3 times, so the output will be [0,0,0]

```
x=[1,2]*3
```

Same as before, but with two elements, the output will be [1,2,1,2,1,2]

```
x=(2)*3
```

Here instead of a list we have a Tupple, so the multiplication instead of repeating the object, it multiplies it, therefore result = 6

```
x=(2,1)*3
```

Same as before, but for each element, result = (6,3)

```
x = map(len, zip([1, 2])); x = print(x)
```

Here we have two parts, let’s organize it a little more.

```
x = map(len, zip([1, 2]));
x = print(x)
```

First of all we made a zip of [1,2] and we gave us a list of two tupples, each one containing one element.
We mapped inside the memory, the function Len and these two tupples, so when calling X, we are calling this function over these tupples of Len 1,
The result will be [1,1]

```
x = zip(*[‘a’ * 3], map(len, zip([’2’] * 3))); x.__next__()
```

Okay, that’s harder… Let’s do it step by step

```
[‘a’ * 3 = [“aaa”]
[“2”] * 3 = [“2”, “2”,”2”]
```

```
x = zip(* [“a”,”a”,”a”] , map(len, zip( [“2”,”2”,”2”];
x.__next__()
```

As before, map function will map the function Len over the result of the zip funcion, giving us the length of each element in the list, therefore:

```
x = zip(* [“aaa”] , [1,1,1]);
x.__next__()
```

What it does the first part, is take each element of the first set, and each element of the second one, and combines them like this:

```
x = [('a', 1), ('a', 1), ('a', 1)]
```

So the last question is: what does x.__next__() do?

Next is used to manage iterators in Python. However, since we do not have any iterator here, it returns “StopIteration”, which means that are no more elements lefts for the iterator to check.

```
x=[x for x in map(lambda x: x**2/2**2,[2,4])]
```

Let’s simplify this:

```
x for x in map(lambda x: x^2/4 ,[2,4])]
```

We are defining a lambda function. This function defines that the output x (at the left) will be the input x (at right) squared and divided by 4.

This function will be mapped to the list on the right and will be called when doing x for x. So our result will be this: [1.0,4.0] (With the division they are converted to floats)