Python programming
Lab. Work n° 4 : Searching

Preliminaries : Please remind that teachers can be called to help you on any problem you get. Don't get stuck on an issue for too long.

Exercise n°1 : Search

1. In the module `myarray.py` add a function `sorted_array(n)` that creates an array of n elements, all stored in ascending order, such that $A[i+1] - A[i] \in [0, 5[$. Test it :

```
Size?10
[2, 5, 6, 10, 15, 17, 21, 25, 27, 30]
```

2. Create a module `search.py`
3. In module `search.py`, write a function `linear_search(a,e)` that returns `True` if element e in in the sorted array a and `False` if not. Test it in a module `search_main.py` :

```
[3, 6, 7, 9, 13, 15, 15, 15, 19, 21, 26, 31]
Value to search? 4
False
[3, 6, 7, 9, 13, 15, 15, 15, 19, 21, 26, 31]
Value to search? 15
True
[3, 6, 7, 9, 13, 15, 15, 15, 19, 21, 26, 31]
Value to search? 3
True
[3, 6, 7, 9, 13, 15, 15, 15, 19, 21, 26, 31]
Value to search? 31
True
[3, 6, 7, 9, 13, 15, 15, 15, 19, 21, 26, 31]
Value to search? -1 ⟵value to stop...
```

4. In module `search.py`, add a function `dichotomic_search(a,e)` that return `True`▷ ▷ if element e in in the sorted array a and `False` if not, using the iterative dichotomic method. Test it in the module `search_main.py` such that both `linear_search` and `dichotomic_search` are computed :

```
[0, 2, 6, 11, 12]
Value to search? 3
Dichotomic=False Linear=False
[0, 2, 6, 11, 12]
Value to search? 5
Dichotomic=False Linear=False
[0, 2, 6, 11, 12]
Value to search? 12
Dichotomic=True Linear=True
[0, 2, 6, 11, 12]
Value to search? -1
```

5. In module `search.py` add a function `recursive_dichotomic_search(a,e)` that computes the same as `linear_search` in a recursive way. Modify `search_main.py` to test it against the two others :

```
[3, 5, 8, 9, 14, 14]
Value to search? 2
Dichotomic=False, Dichotomic (recursive)=False Linear=False
[3, 5, 8, 9, 14, 14]
Value to search? 9
Dichotomic=True, Dichotomic (recursive)=True Linear=5
[3, 5, 8, 9, 14, 14]
Value to search? -1
```

6. In `search_main.py` add a function `test` that tests if all three searching algorithms have consistent results over N sorted arrays $\{A_{1\leqslant s\leqslant N}\}$, $\forall s \in [1, N], |A_i| = s$. For each A_k, you may test consistency for all integer values in $[A_k[0] - 1, A[k-1] + 1]$. If results are not consistent, then you need to find the bug !

7. Modify all of these such that you will be able to collect the number of comparisons made in the mean for all three algorithms and draw the corresponding functions (compare with $fx() = x$ and $g(x) = \log_2 x$) :