Software Engineer Python Test

Instructions to candidates:

- You are to complete this test within 1 hour.
- This is an open book test (Google probably won't help you much though).
- If any of the questions are unclear, state clearly on what your assumption is and work on the question with those assumptions.
- Non-coding questions should have short and sweet answers. Essays are not allowed.
- If any non-coding answer (with its assumption) is longer than 40 words, it will be ignored and 0 marks will be awarded for that question.
- Coding questions should be answered as pedantically as possible. The code should be optimized as much as you can and it must run.

Submission Guidelines:

- Zip this document along with all additional files and send it back to us. Ensure that the zip file is renamed as such: <first_name>_<last_name>.zip

Graders Table

Section	Score
The Sequence	/4
Has Duplicate	/3
Mystery function	/6
The Syracuse function	/7
Total	/20

1.0 - Python Test

1.1 – The Sequence

We consider the sequence defined as:

$$\begin{cases} u_0 = 2 \\ u_n = 3 * u_{n-1} - 1 \end{cases}$$

1. Write an iterative function that takes as parameter a natural integer n and computes the term u_n of the sequence by using a while loop.
   (1 point)

2. Write an iterative function that takes as parameter a natural integer n and computes the term u_n of the sequence by using a for loop.
   (1 point)

3. Write an recursive function that takes as parameter a natural integer n and computes the term u_n of the sequence.
   (1 point)

4. Write a function which, given an integer m, calculates the index of the first term of the sequence that is greater than or equal to m (example: if m = 30, the function will return 3 because all terms of index less than 3 are smaller than 30).
   (1 point)

1.2 – Has Duplicate

You are to create 3 functions in section(s) ***a***, ***b***, and ***c***.
Given the following code snippet:

a = [1,4,3,8,16]
b = [1,0,0]
c = [1,0,2,3]

1. hasDuplicate (1 point)

Write a function hasDuplicate(t) which return True if the table has duplicate and False if the table has no duplicates. Explain and justify the complexity of this function.
Expected output(s):
hasDuplicate(a): False
hasDuplicate(b): True
hasDuplicate(c): False

2.Internal (1 point)

Write a function internal that return True if $\forall i \in [0, n], t[i] \in [0, n]$ and False otherwise.

Expected output(s):

internal(a): False
internal(b): True
internal(c): True

3. permutation (1 point)

The table t is a permutation if $\forall i \in [0, n], t[i] \in [0, n]$ **and** the table has no duplicates. Write the function permutation(t)

Expected output(s):

permutation(a): false
permutation(b): false
permutation(c): true

1.3 – Mystery function

Mystery function

```
def mystery(t,k)
        if k == len(t) -1:
                return True
        if t[k] > t[k+1]:
                return False
        return mystery (t,k+1)
```

1. t=[6,9,4,8,12], What is the return value for mystery(t,2), (describe all the recursive calls) **(1 point)**

2. What is the return value for mystery(t,0), (describe all the recursive calls) **(1 point)**

3. What is the function mystery doing, could you please describe the information that it provide. **(1 point)**

4. What is the maximum number of recursive calls if the table size is n. **(1 point)**

5. Write a function isGrowing(t) using the function mystery**(1 point)**

6. Write a recursive function isInside(t,x,k) which return True is x is in the table t starting from index k, False otherwise. **(1 point)**

1.4 – The Syracuse function

The function syr(n) is defined, n is an integer:

```
def syr(n):
    while n>1:
        n=n//2
    return n
```

1. Write the result for the two values: 32, 20. Please give all the successive values for n **(1 point)**

2. Describe the function complexity in the best case and the worst case **(1 point)**

The function syrac(n) is defined, n is an integer:

```
def syrac(n):
    while n>1:
        if n%2 == 0:
            n=n//2
        else :
            n =1
    return n
```

3. Write the result for the two values: 32, 20. Please give all the successive values for n. **(1 point)**

4. Describe the function complexity in the best case and the worst case, when the best case and the worst case are happening, can you describe precisely the cases. **(1 point)**

The function syracuse(n) is defined, n is an integer:

```
def syrac(n):
    while n>1:
        if n%2 == 0:
            n=n//2
        else
            n=3*n +1
    return n
```

5. Write the result for the two values: 32, 20. Please give all the successive values for n.

Since 1937, mathematicians and computer scientists have been studying the Syracuse Sequence. They want to know whether these sequences are all finite, or whether there are cyclical ones, or whether they exist which contain a strictly increasing subsequence. This problem is an open problem, which means that no one knows the answer. **(1 point)**

6. Describe the function complexity in the best case and the worst case, when the best case and the worst case are happening, can you describe precisely the cases.
(2 points)