Ejercicio 3

3. By estimating the number of operations as a function of n, estimate the complexity of the following function

```
def func1(n):
    A = range(0,n)
    sum = 0
    i = 0
    for x in A[i:]:                    n = O(n)
        i += 1
        for j in range(i,len(A)):      Σ_{i=2}^{n} i = n(n-1)/2 = O(n^2)
            y = A[j]
            k = j
            while k<len(A)             O(log(n))
                z = A[k]
                k = 2 * k
                if x+y <= z
                    sum += 1
    return sum
```

Por tanto:

$O(\text{func1}(n)) = n \cdot n^2 \cdot \log(n) = n^3 \log(n)$