Q2
You are given pseudocode of an algorithm. Write the formula for the output of this algorithm in terms of the input n(write the output a in terms of input n). Find the worst-case time complexity of the algorithm by showing all the steps.

```
procedure nestedloops(n)
        a←0
        for i ← 1 to n do
                for j←1 to i^2 do
                        for k ← 1 to i do
                                a←a+1
                        end for
                end for
        end for
        return a
end procedure
```

Output formula:

Since all we have is for loops, we can directly write the output in terms of summation. a is increased by 1 so we sum 1's inside the summation. Also, we can expand the summation as following:

$$a = \sum_{i=1}^{n} \sum_{j=1}^{i^2} \sum_{k=1}^{i} 1 = \ 1^2 * 1 + \ 2^2 * 2 + \cdots + n^2 * n = \ \sum_{i=1}^{n} i^3$$

This is equal to sum of cubes of numbers from 1 to n. So the output formula becomes:

$$a = \frac{n^2\ (n+1)^2}{4}$$

Worst case complexity

Basic operation in this function is a←a+1. So, we need to find how many times this expression executed. Since a starts from 0 and every time this expression is executed it is increased by one, previous formula we found equals to number of times this operation is executed. Since the code only consists of nonconditional loops we always get the same number for given input size (worst case complexity).

Output = execution count = $f(n) = \frac{n^2\ (n+1)^2}{4} = \frac{n^4}{4} + \frac{2n^3}{4} + \frac{n^2}{4}$

When we ignore lower order terms and constants we get

$$W(n) \in O(n^4)$$