Translating mathematical functions to code with time asymptotic analysis

---

Suppose the following for some finite k, $M = \{m_1, m_2, \ldots, m_k\} \wedge N = \{n_1, n_2, \ldots, n_k\}$:

$$f(k) = \prod_{i=1}^{k} \sum_{j=1}^{k} \left(m_i \times n_j\right)$$

The given **function** can be similarly expressed as:

$$f(k) = \left(\sum_{j=1}^{k} m_1 \times n_j\right) \times \left(\sum_{j=1}^{k} m_2 \times n_j\right) \times \cdots \times \left(\sum_{j=1}^{k} m_{k-1} \times n_j\right) \times \left(\sum_{j=1}^{k} m_k \times n_j\right)$$

Or even more explicitly:

$$f(k) = \Big[(m_1 \times n_1) + (m_1 \times n_2) + \cdots + (m_1 \times n_j)] + \cdots + [(m_i \times n_1) + (m_i \times n_2) + \cdots + (m_i \times n_j)$$

Code translation in `Python3`

The proposed function $f(k)$ is implemented below.

```
"""
Code translation in Python3
Suppose M, N are lists of some fixed size k
"""

product: int = 1
for m_i in M:
        sum_buffer: int = 0.
        for n_j in N:
                sum_buffer += (m_i * n_j)
        product *= sum_buffer
```

Code analysis

Let `product` be 1 in the initial state, that is, for some value v_1 computed in the first iteration of the outer `for` loop holds: `product` $= v_1$. Thus, `product` $= \prod_{i=0}^{k} v_i$ for k-th iteration. Furthermore, let `sum_buffer` (previously indicated as v_i) be initially 0 in each iteration of the outer `for` loop, so for each iteration in the inner `for` loop, `sum_buffer` equals to the sum of all terms of N, namely n_j, multiplied by some current constant m_c, thus `sum_buffer` $= \sum_{j=1}^{k} m_c \times n_j$, where c is a constant and $c \leq k$.

Time complexity

According to **asymptotic analysis**, the lower and the upper bounds of such algorithm remain **identical**, and, undeniably, any case holds the same running time complexity. Thus, we compute the average (and, frankly, the only possible) running time as $\Theta(k^2)$ where the only variable affecting the real running speed is the value of k. Assume the parabolic function $g(k) = k^2$, where k is an integer and $k \geq 1$.