CSCE 221
Problem Set 3

Jacob Purcell, Texas A&M, Student

I. Problem 1:

A:

Described on the figures below are O values next to their corresponding computations. $Fragment_1$ is found to have the behavior of $O(N)$, $Fragment_1$ goes as $O(N^2)$, and finally $Fragment_1$ goes as $O(N^4)$. In each case, the separated O values are considered against eachother as $N \to \infty$. Since the functions separated in the figures are connected linearly, as $N \to \infty$ the influence of smaller O becomes negligible.

$$Fragment_1 : \boxed{O(N)}$$

$$Fragment_2 : \boxed{O(N^2)}$$

$$Fragment_3 : \boxed{O(N^4)}$$

```
sum = 0                 O(0)
for i from 1 to n do    O(N)
    sum = sum + 1
end                     O(0)
```

Fig. 1. $Fragment_1$

```
sum = 0                    O(0)
for i from 1 to n do       O(N)
    for j from 1 to i do
        sum = sum + 1      O(n*i)
    end                    O(0)
end
```

Fig. 2. $Fragment_1$

```
sum = 0                          O(0)
for i from 1 to n do             O(N)
    for j from 1 to i^2 do       O(n*i^2)
        for k from 1 to j do     O(n*i^2*j)
            sum = sum + 1        O(1)
        end
    end                          O(0)
end
```

Fig. 3. $Fragment_1$

B:

Code attached in separate document.

C:

Figure 4. shows plotted output of the functions' recorded time growth with respect to the number of operations. The asymptotic trends in each fragment can be seen by dividing the times with their counterparts. As it turns out, the estimates approached a constant in separation for each recorded function. This was verified by dividing several points along the curves, getting a similar constant every check. As well, the constant would gradually approach what is to be construed as the "true" proportionality constant. Following the form

$$Fragment_n(N) = \gamma O(N), \; \gamma = \frac{Fragment_n(N)}{O(N)}$$

where γ is the constant of proportionality and $Fragment_n$ is the function of time, using 1000 datapoints, γ has been found to be:

$$Fragment_1 : \; 1$$

$$Fragment_2 : \; 0.5005$$

$$Fragment_3 : \; Diverges$$

Running $Fragment_3$ to convergence proved difficult, and after 12 hours the following result was attained. $\gamma_{250} - \gamma_{240} = 1, \gamma_{377} - \gamma_{367} = 1$. This shows that the distance between proportionality constants every 10 iterations grows by 1. In order to show convergence, this difference must approach zero. Since γ diverges, $Fragment_3 > O(N^4)$. A guess was made and convergence was found when compared to $O(N^5)$. Upon further investigation, a mistake in the initial estimation was found. Although "$i * j * k$" is correct, we must break the iterators down into their corresponding factors of $N, i \to N, j \to i * i \to N^2, k \to j \to N^2, \therefore i * j * k = N^5$, thus we find γ of $Fragment_3$ to be 0.1005.

$$Fragment_1 : \boxed{O(N)}$$

$$Fragment_2 : \boxed{O(N^2)}$$

$$Fragment_3 : \boxed{O(N^5)}$$

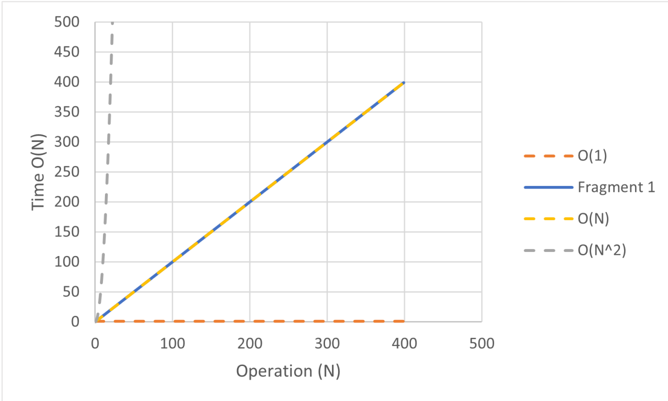

Fig. 4. $Fragment_1, O(N)$

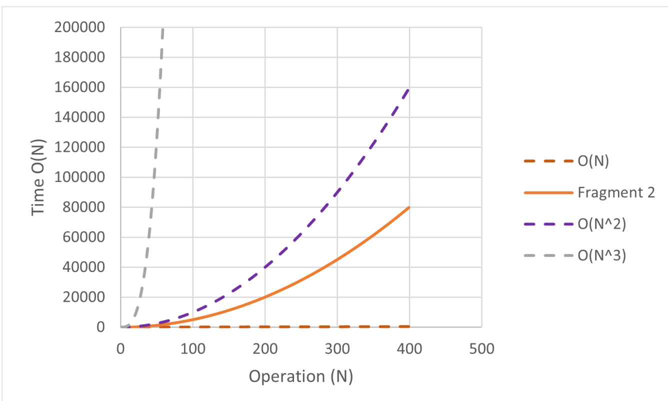

Fig. 5. $Fragment_2, O(N^2)$

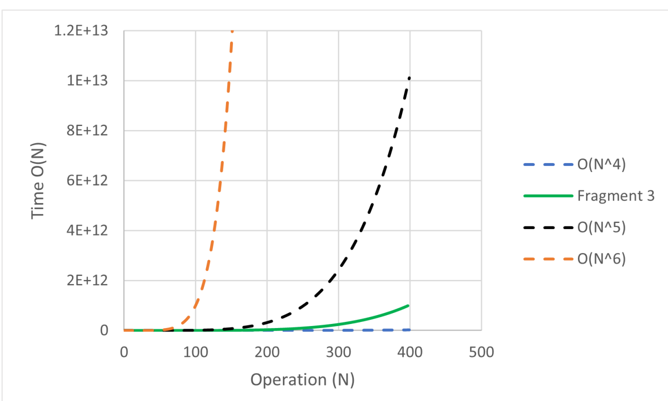

Fig. 6. $Fragment_3, O(N^5)$

II. Problem 2:

A:

Psuedocode for hand calculations, A + B:
Since addition is commutative, the larger number will be called A.

For each digit in A that has a pair with B, $O(N)$
 add, $O(1)$
 store first digit, $O(1)$
 carry the 1 if present, $O(1)$
end

$$\boxed{O(N)}$$

B:

Psuedocode for hand calculations, A * B:
Since multiplication is commutative, the larger number will be called A.

For each digit in A that has a pair with B, $O(N)$
 Multiply, $O(1)$
 store first digit, $O(1)$
 multiply the exess by number of zeroes in the B digit, $O(N)$
 add excess to next iteration, $O(1)$
end

$$\boxed{O(N^2)}$$

C:

Psuedocode for hand calculations, A / B:

For each digit in A, $O(N)$
 divide the front digit of A by the front digit in B, $O(1)$
 store first digit (appended to rightmost end of number), $O(1)$
 store A % B, $O(1)$
 subtract A % B from the next iteration, $O(1)$
end

$$\boxed{O(N)}$$

III. Problem 3:

$$Given : f(x) = \Sigma_{i=0}^{N} a_i x^i$$

A:

Naive exponentiation is estimated to be $\boxed{O(N^2)}$ since not only do summations grow with N, but x^N takes N multiplications to calculate.

B:

Fast exponentiation follows the system:

$$X^N = X^{2^{\frac{N}{2}}} \; for\ even$$

$$X^N = X * X^{2^{\frac{N-1}{2}}} \; for\ odd$$

Considering the even case, $X^{2^{2^{\cdot^{\cdot^{\cdot^{\frac{N}{2^k}}}}}}}$. Requires k operations ($O(k)$), which if we continue until $\frac{N}{2^k} = 1$, then

$$N = 2^k$$

$$log_2 N = k$$

So, $O(k) \rightarrow O(\log_2 N)$, summed N times, which makes $f(x)$ bounded by $\boxed{O(N \log_2 N)}$.

C:

Given function is estimated to be $O(N)$, since it computes 3 operations N times.

```
value = 0
for i from n to 0 by -1 do
    value = value * x + a[i]
end
```

Fig. 7.

IV. Puzzle:

Show that X^{64} can be calculated in 8 multiplications.

$$X * X = X^2 \tag{1}$$

reusing X^2 and continuing the pattern leads to

$$X^2 * X^2 = X^4 \tag{2}$$

$$X^4 * X^4 = X^8 \tag{3}$$

$$X^8 * X^8 = X^{16} \tag{4}$$

$$X^{16} * X^{16} = X^{32} \tag{5}$$

$$X^{32} * X^{32} = X^{64} \tag{6}$$

6 multiplications using this method.