QUESTION 2, PART B

length N

GOAL: Calculate linear convolution between $y[k]$ & $x[k]$ as $y[k]*x[k]$

or in frequency domain. $IDFT\{DFT\{x[k]\ 0\ 0 \ldots 0\} \cdot DFT\{y[k]\ 0\ 0 \ldots 0\}\}$.

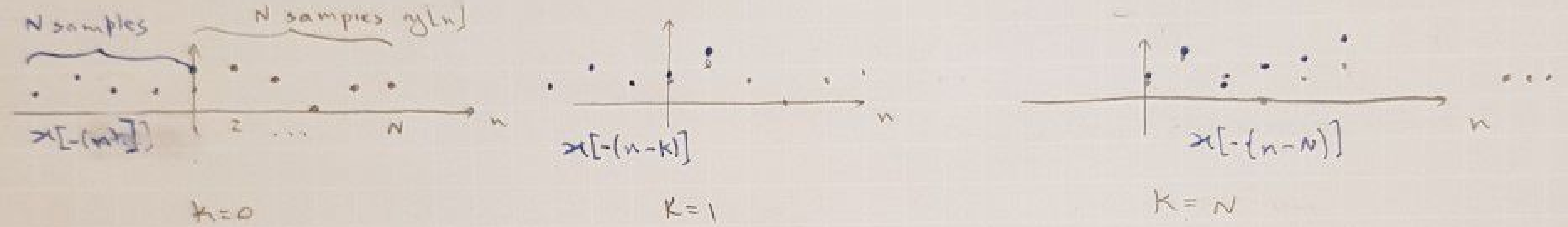
Plot number of multiplications needed in terms of N.

SOLUTION:

(i) Time-domain:

$$z[k] = y[k]*x[k] = \sum_{n=-\infty}^{\infty} y[n] \cdot x[k-n] = \sum_{n=-\infty}^{\infty} y[n] \cdot x[-(n-k)] \quad (\text{where } y[n], x[n] \in \mathbb{R}\ \forall n).$$

The number of non-zero multiplications necessary to calculate $y[k]*x[k]$ is:



1 multiplication $\qquad$ 2 multiplications $\qquad \ldots \qquad$ N multiplications

$$(1+2+\ldots+N-1)+N+(N-1+\ldots+2+1) = \underset{\substack{\uparrow \\ \text{arithmetic sum}}}{N} + \cancel{2} \cdot \frac{(\cancel{1}+N-\cancel{1}) \cdot (N-1)}{\cancel{2}} = N + N^2 - N = N^2$$

To get the result above we just count the multiplications from the intersection of $y[n]$ and $x[-(n-k)]$, $k = \{0, 1, \ldots, 2N\}$, starting from the first point they intersect ($k=0$) until the last ($k = 2N$).

Other assumptions: $x[k], y[k] = 0\ \forall\ k < 0$.

(ii) Frequency domain

$$IDFT\{DFT\{x[n]\ 0\ 0 \ldots 0\} \cdot DFT\{y[n]\ 0\ 0 \ldots 0\}\}$$

According to page 52 in the book, each FFT algorithm requires $N \cdot \log_2(N)$ multiplications (# samples in signal)

In order to make circular convolution coincide with linear convolution, pad both x and y with N zeros each. Then the number of multiplications is:

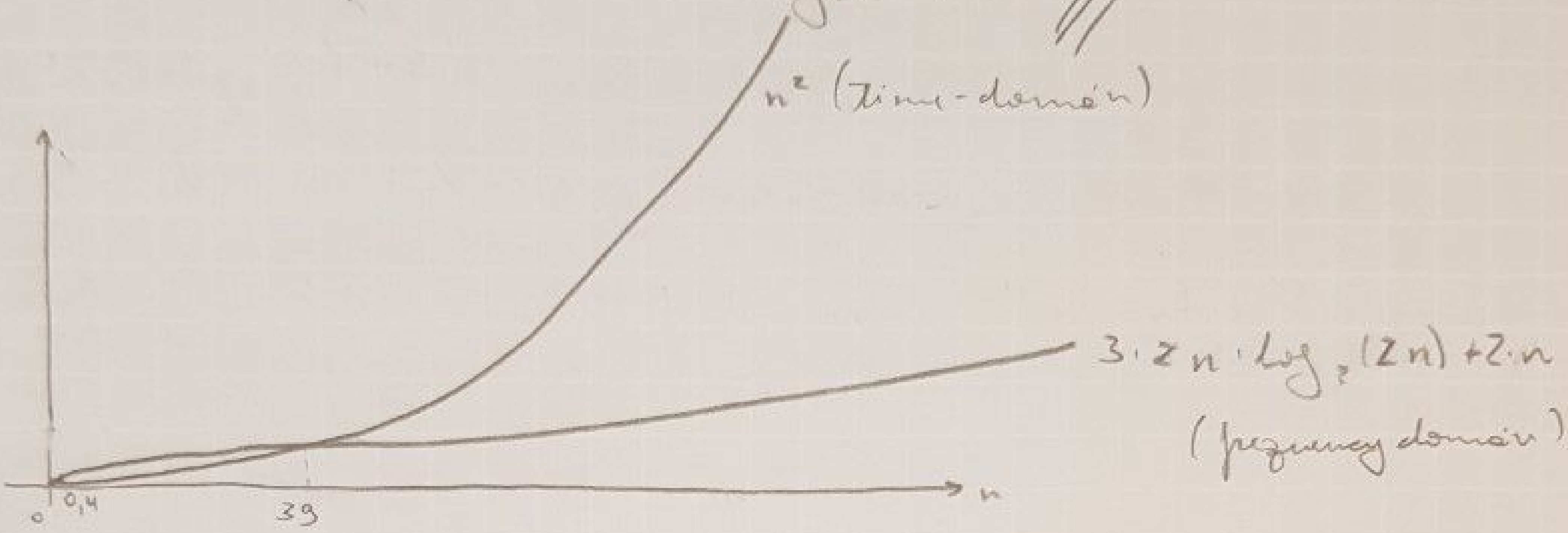
- $DFT\{\underbrace{\widehat{x[n]\ 0\ 0 \ldots 0}}_{2N\text{-long signal}}\} \longrightarrow (2N) \cdot \log_2(2N)$ multiplications
- $DFT\{\widehat{y[n]\ 0\ 0 \ldots 0}\} \longrightarrow (2N) \cdot \log_2(2N)$ multiplications
- $DFT\{x[n]\ 0\ 0 \ldots 0\} \cdot DFT\{y[n]\ 0\ 0 \ldots 0\} \longrightarrow 2N$ multiplications (continues $\rightarrow$)

(continuation)

$$\cdot \ IDFT\{\overbrace{DFT\{x[n]\ 0\ 0 \ldots 0\} \cdot DFT\{y[n]\ 0\ 0 \ldots 0\}}^{2N \cdot \log}\} \longrightarrow (2N) \cdot \log_2(2N)$$

Therefore the total multiplications is: $3 \cdot 2N \cdot \log_2(2N) + 2N$ //

(iii) Plot:



This plot was generated with the help of Matlab:

```
syms x;
fplot(x^2); hold on;
fplot(6.x.log(2x)+2x)
```