**MUSI-7100 Timeline**

Spring 2019

Punahamoa Walker ＆ Yi Wu

**Question 1) Time Domain Convolution:**

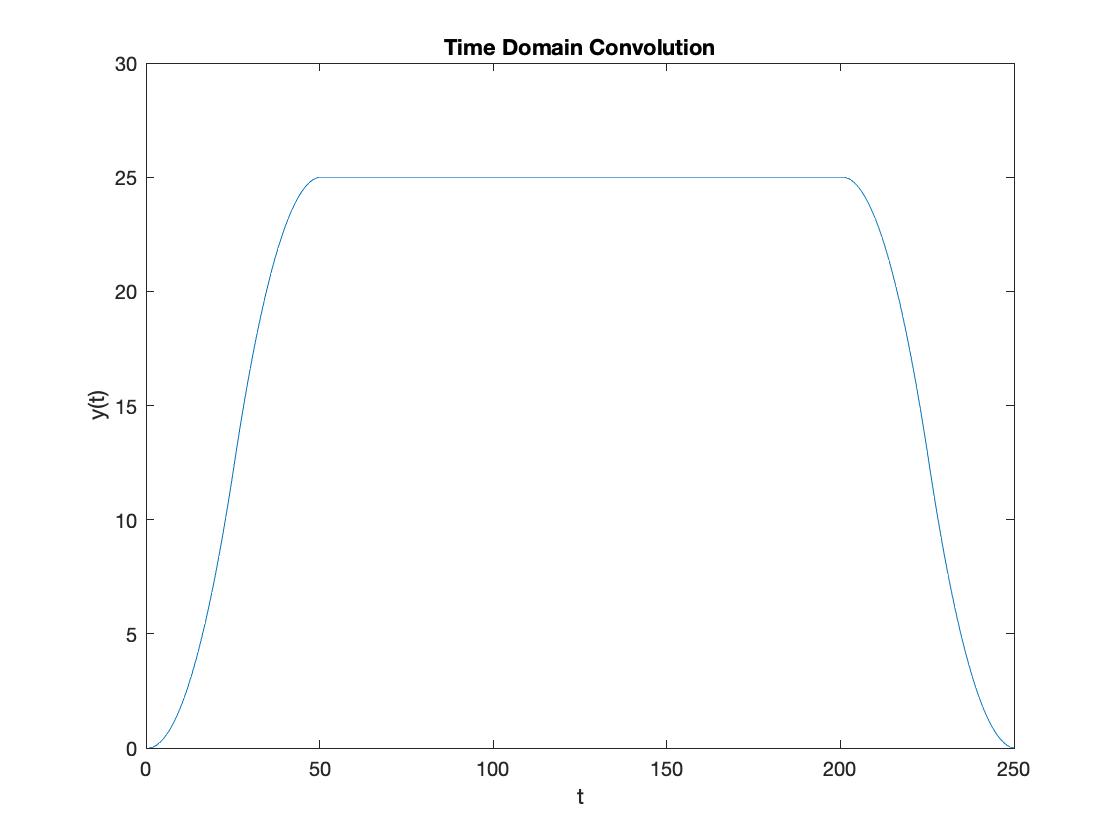
**If the length of 'x' is 200 and the length of 'h' is 100, what is the length of 'y' ?**

**Answer**:

**'x': DC signal of length 200**

**'h': symmetric triangular signal of length 51**

**Plot**:



**Figure. 1 Time Domain Convolution**

**Question 2) Frequency Domain Convolution:**

**Zero-padding the signals appropriately before the transform?**

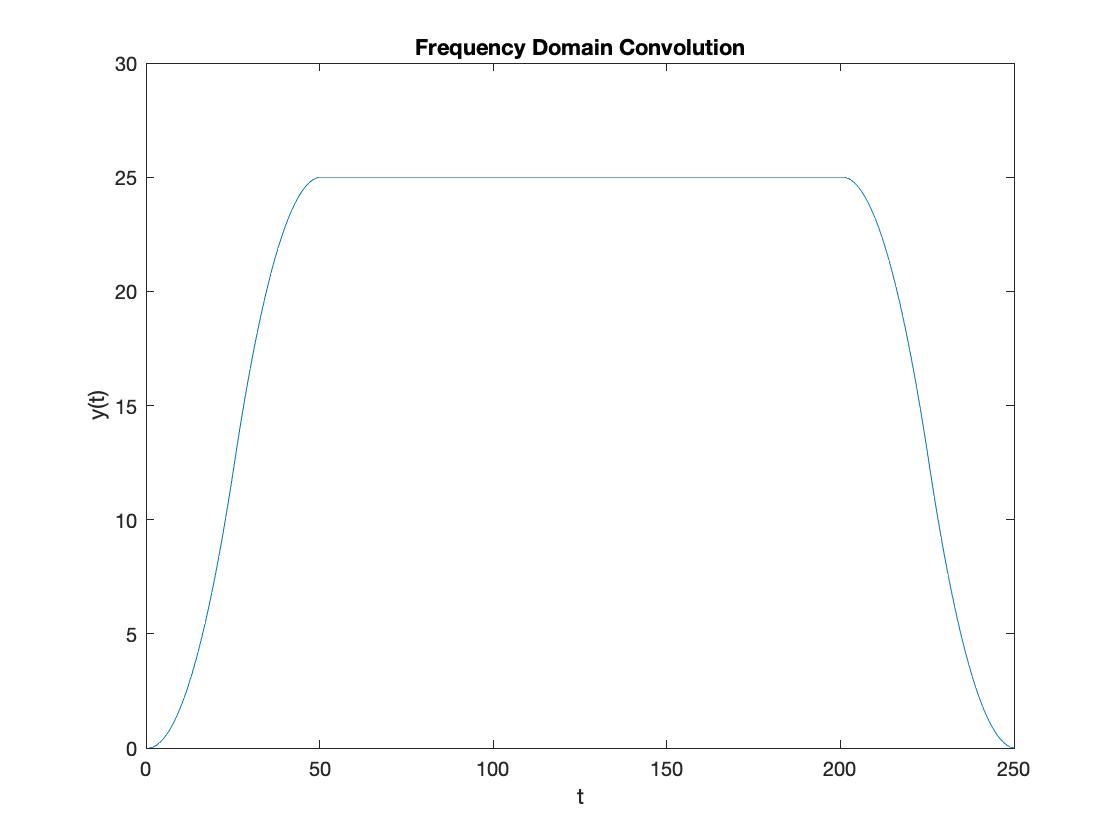
**Answer:**

Zero-padding is to avoid mixing convolution results due to circular convolution. Since the length of the convolution output is longer than two input vectors and thus without zero-padding will result in FFT to mix up the output.

**'x': DC signal of length 200**

**'h': symmetric triangular signal of length 51**

**Plot**:



**Figure. 2 Frequency Domain Convolution**

**Question 3) Compare with MATLAB Implementation:**

**‘x’: DC signal of length 200**

**‘h’: symmetric triangular signal of length 51**

|  |  |
| --- | --- |
| **Method** | **Time(s)** |
| conv() |  |
| myTimeConv() |  |
| myFreqConv() |  |

**Table. 1 Time Comparison**

|  |  |  |
| --- | --- | --- |
| **Comparison** | **y\_time** | **y\_freq** |
| Mean Difference with conv() |  |  |
| Mean Absolute difference with conv() |  |  |
| Standard Deviation of the Difference with conv() |  |  |

**Table. 2 Output Difference with conv() Statistics**

**‘x’: ‘piano.wav’**

**‘h’: ‘impulse-resopnse.wav’**

|  |  |
| --- | --- |
| **Method** | **Time(s)** |
| conv() |  |
| myTimeConv() |  |
| myFreqConv() |  |

**Table. 3 Time Comparison**

|  |  |  |
| --- | --- | --- |
| **Comparison** | **y\_time** | **y\_freq** |
| Mean Difference with conv() |  |  |
| Mean Absolute difference with conv() |  |  |
| Standard Deviation of the Difference with conv() |  |  |

**Table. 4 Output Difference with conv() Statistics**

**Which is the most time-efficient method and which is the least time-efficient method? Discuss the possible reasons.**

Answers: From the time comparison table 1 and table 3, we can see that time convolution method takes the longest time to get result and it is the least time-efficient method. This is due to the for loop in the algorithm takes longer time (matlab are more efficient with vector computation.) Matlab built in method and our frequency domain convolution method come very close and is way better than the time convolution method (our frequency convolution method is slightly more efficient.) This is because we taking the advantage of vector computation instead of using for loop.