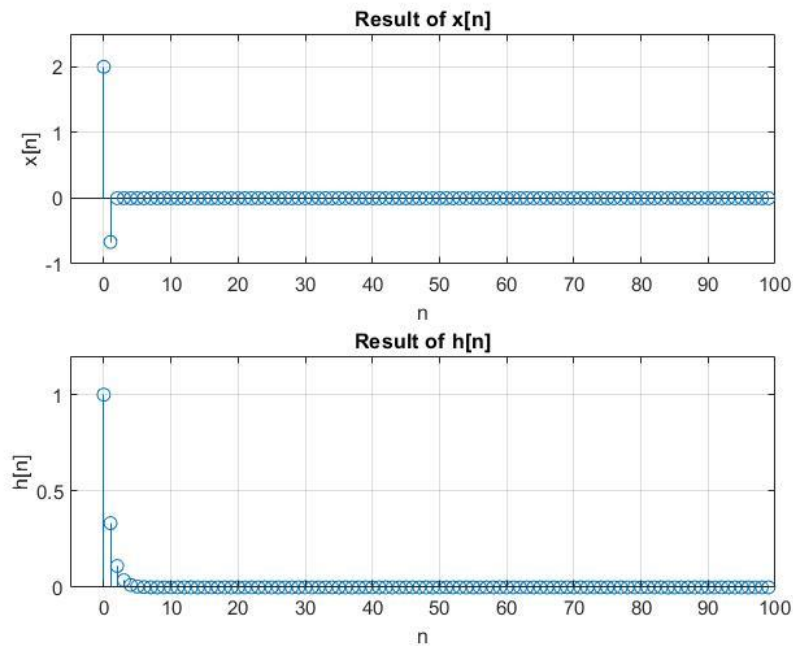
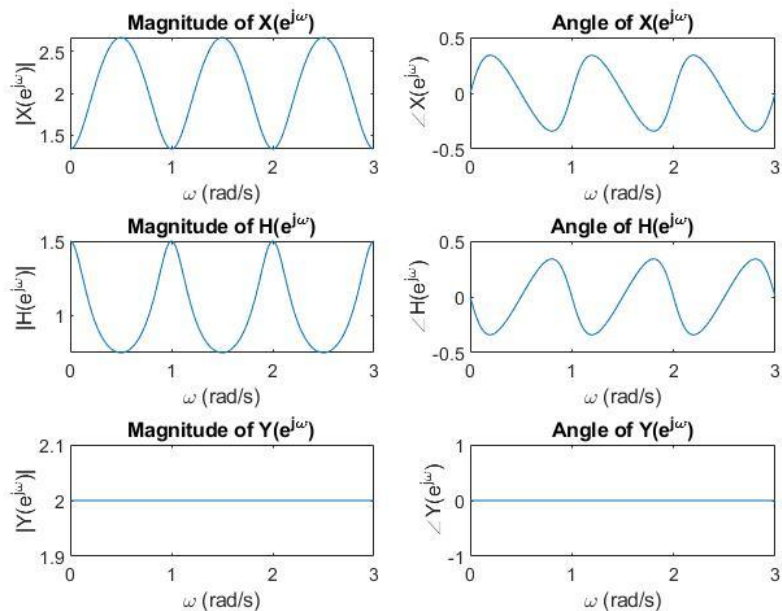


3(a)



首先 $x[n]$ 的值先填入 2 與 $-2/3$ ，後續全部填 0。接著 $h[n]$ 的部分因為要觀察的部分正好大於 0，因此將正確的正整數 n 值代入 $(1/3)$ 的次方項就可以達成。接著使用 stem 將結果畫出，結果如上。

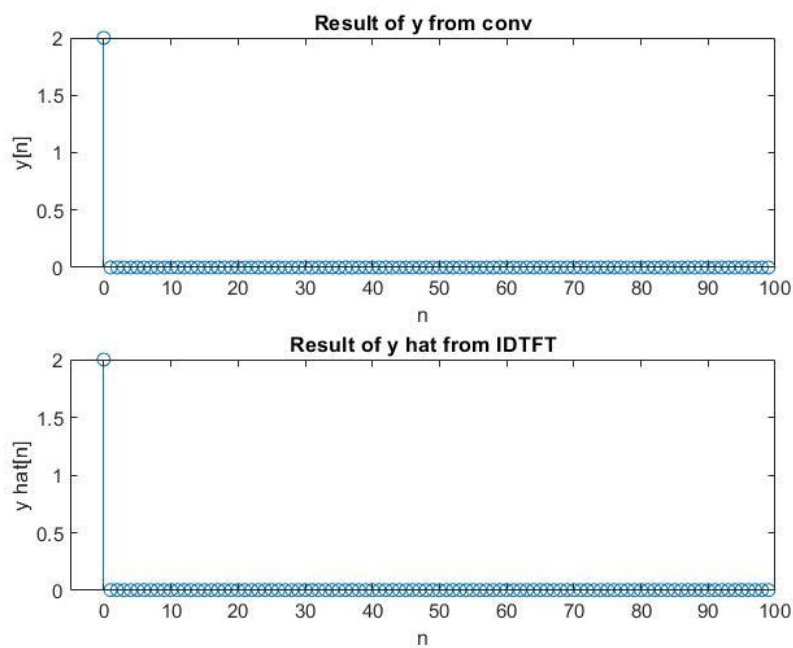
3(b,c)



首先使用定義來獲得 $X(e^{j\omega})$ 與 $H(e^{j\omega})$ 。x 和 n 為 1×100 的矩陣， ω 為 1×601 的矩陣，如果要將矩陣相乘，則必須進行維度調整。首先在於 exp 項，n 為 1×100 、 ω

為 1×601 的矩陣，為了之後能和 x 矩陣相乘，此處必須先變成 $100 \times __$ 的矩陣。由此可知，必須要將 n 先轉置後跟 ω 相乘，得到 100×601 的矩陣， x 才有辦法乘以此矩陣得到最後的 1×601 的矩陣 $X(e^{j\omega})$ ，而 $H(e^{j\omega})$ 也是使用相同的方法。 $Y(e^{j\omega})$ 的部分由於目前在頻域，直接將 $X(e^{j\omega})$ 與 $H(e^{j\omega})$ 相乘就是結果。最後將三者的大小與相位畫出後，結果如上圖。

3(d)



首先使用 `conv` 來算出 y ，使利用了 MATLAB 本身的函式庫，將 x 與 h 代入後可以直接算出 y 的時域訊號。而 y_hat 則是先將 x 與 h 丟到頻域後得到 $X(e^{j\omega})$ 與 $H(e^{j\omega})$ ，將兩者相乘後經由 IDTFT 算出 y 的時域訊號。由上圖可以發現，兩者算出的結果是完全一樣的。