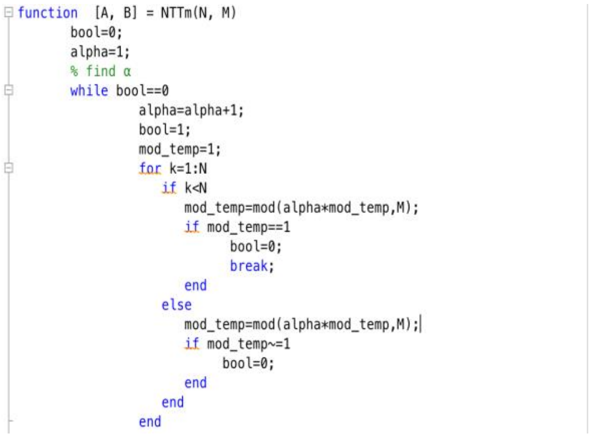
ADSP_HW5

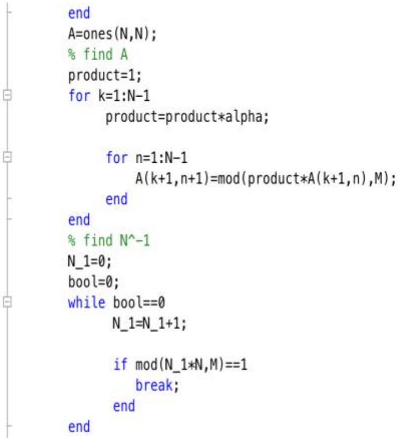
電機碩一　r06921048　李友岐

(1)

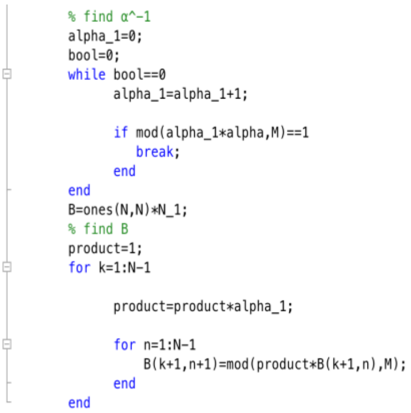
1. matlab code:



```
function  [A, B] = NTTm(N, M)
        bool=0;
        alpha=1;
        % find α
        while bool==0
                  alpha=alpha+1;
                  bool=1;
                  mod_temp=1;
                  for k=1:N
                     if k<N
                        mod_temp=mod(alpha*mod_temp,M);
                        if mod_temp==1
                              bool=0;
                              break;
                        end
                     else
                        mod_temp=mod(alpha*mod_temp,M);
                        if mod_temp~=1
                             bool=0;
                        end
                     end
                  end
```

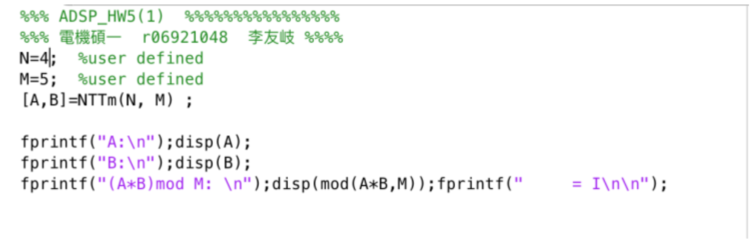


```
        end
        A=ones(N,N);
        % find A
        product=1;
        for k=1:N-1
             product=product*alpha;

             for n=1:N-1
                 A(k+1,n+1)=mod(product*A(k+1,n),M);
             end
        end
        % find N^-1
        N_1=0;
        bool=0;
        while bool==0
              N_1=N_1+1;

              if mod(N_1*N,M)==1
                 break;
              end
        end
```

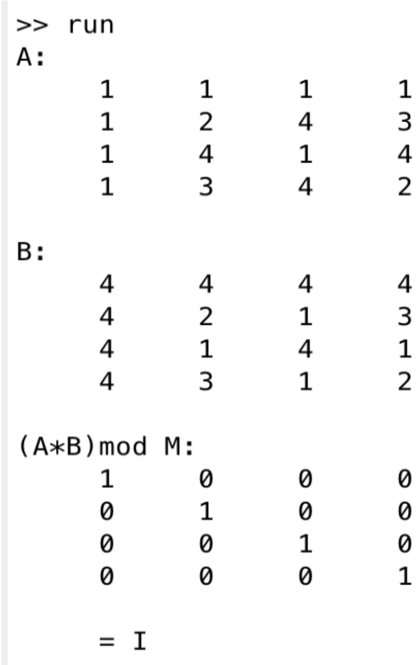


```
        % find α^-1
        alpha_1=0;
        bool=0;
        while bool==0
              alpha_1=alpha_1+1;

              if mod(alpha_1*alpha,M)==1
                 break;
              end
        end
        B=ones(N,N)*N_1;
        % find B
        product=1;
        for k=1:N-1

              product=product*alpha_1;

              for n=1:N-1
                  B(k+1,n+1)=mod(product*B(k+1,n),M);
              end
        end
```

2. sample result



```
%%% ADSP_HW5(1)  %%%%%%%%%%%%%%%%
%%% 電機碩一  r06921048  李友岐 %%%%
N=4;  %user defined
M=5;  %user defined
[A,B]=NTTm(N, M) ;

fprintf("A:\n");disp(A);
fprintf("B:\n");disp(B);
fprintf("(A*B)mod M: \n");disp(mod(A*B,M));fprintf("     = I\n\n");
```



```
>> run
A:
     1     1     1     1
     1     2     4     3
     1     4     1     4
     1     3     4     2

B:
     4     4     4     4
     4     2     1     3
     4     1     4     1
     4     3     1     2

(A*B)mod M:
     1     0     0     0
     0     1     0     0
     0     0     1     0
     0     0     0     1

     = I
```

(2)

1. The complexity of computation is in linear time.

2. The hardware architecture is fixed for different input signal length.

(3)

(a) $W_2 = 2$

$W_4 = 2{*}W_2 + 4 = 8$

$W_8 = 2{*}W_4 + 8 = 24$

$W_{16} = 2{*}W_8 + 16 = \underline{64}$

$W_{32} = 2{*}W_{16} + 32 = \underline{160}$

(b) $h[6] = f[0]g[6\oplus0] + f[1]g[6\oplus1] + f[2]g[6\oplus2] + f[3]g[6\oplus3] + f[4]g[6\oplus4] + f[5]g[6\oplus5] + f[6]g[6\oplus6] + f[7]g[6\oplus7]$

$= f[0]g[6] + f[1]g[7] + f[2]g[4] + f[3]g[5] + f[4]g[2] + f[5]g[3] + f[6]g[0] + f[7]g[1]$

(4)

(a) CDMA (code division multiple access)

(b) Adaboost face detection (extract local feature)

(5)

(c) Integer LTI system analysis:

Since NTT is appropriate for convolution, and it is in integer field.

(d) Encryption:

Since it is difficult to predict the mapping of NTT and the computation for modern cryptosystem can be speeded up by NTT.

(6)

(1) OFDM is orthogonal. There is no interference between different channels.

$A^T = A^{-1}$ → $AA^T = A^TA = I$.

(2) OFDM has fast algorithm, which is similar to DFT.

Since it involves IFFT-FFT operations, it is simple for us to implement it.

(7)

(a)

1st column: [1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1] = x

6th column: [1, 1, -1, -1, -1, -1, 1, 1, -1, -1, 1, 1, 1, 1, -1, -1] = y

11th column: [1, -1, -1, 1, -1, 1, 1, -1, -1, 1, 1, -1, 1, -1, -1, 1] = z

[1,1,0] →[1, 1, -1] modulated by 1st column (x) → [x, x, -x]

[0,1,1] → [-1, 1, 1] modulated by 6th column (y) →[-y, y, y]

[1,0,0] →[1, -1, -1] modulated by 11th column (z) →[z, -z, -z]

We sum 3 channels → [x-y+z, x+y-z, -x+y-z]

and the result of CDMA is as follow

[1, -1, 1, 3, 1, 3, 1, -1, 1, 3, 1, -1, 1, -1, 1, 3,
1, 3, 1, -1, 1, -1, 1, 3, 1, -1, 1, 3, 1, 3, 1, -1,
-1, 1, -1, -3, -1, -3, -1, 1, -1, -3, -1, 1, -1, 1, -1, -3]

(b) No, it is not better.

Since the computation of NTT is much more complicated.